REPORT DOCUMENTATION PAGE

Form Approved OMB NO. 0704-0188

The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggesstions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA, 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any oenalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.
PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE (DD-MM-YYYY)
17-11-2015

2. REPORT TYPE
Ph.D. Dissertation

3. DATES COVERED (From - To)
-

4. TITLE AND SUBTITLE
Computing Distrust in Social Media

5a. CONTRACT NUMBER
W911NF-11-1-0517

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
611102

6. AUTHORS
Jiliang Tang

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES AND ADDRESSES
Arizona State University
ORSPA
AZ Board of Regents on behalf of Arizona State Unive
Tempe, AZ 85287 -6011

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS (ES)
U.S. Army Research Office
P.O. Box 12211
Research Triangle Park, NC 27709-2211

10. SPONSOR/MONITOR'S ACRONYM(S)
ARO

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
60361-LS.33

12. DISTRIBUTION AVAILIBILITY STATEMENT
Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES
The views, opinions and/or findings contained in this report are those of the author(s) and should not contrued as an official Department of the Army position, policy or decision, unless so designated by other documentation.

14. ABSTRACT

A myriad of social media services are emerging in recent years that allow people to communicate and express themselves conveniently and easily. The pervasive use of social media generates massive data at an unprecedented rate. It becomes increasingly difficult for online users to find relevant information or, in other words, exacerbates the information overload problem. Meanwhile, users in social media can be both passive content consumers and active content producers, causing the quality of user-generated content can vary dramatically from excellence to abuse or spam, which results in a problem of information credibility. Trust, providing evidence about with whom

15. SUBJECT TERMS
Distrust, Negative Links, Predicting Distrust, Signed Social Networks, Understanding distrust

16. SECURITY CLASSIFICATION OF:
a. REPORT UU
b. ABSTRACT UU
c. THIS PAGE UU

17. LIMITATION OF ABSTRACT
UU

15. NUMBER OF PAGES

19a. NAME OF RESPONSIBLE PERSON
Huan Liu

19b. TELEPHONE NUMBER
480-727-7349

Report Title

Computing Distrust in Social Media

ABSTRACT

A myriad of social media services are emerging in recent years that allow people to communicate and express themselves conveniently and easily. The pervasive use of social media generates massive data at an unprecedented rate. It becomes increasingly difficult for online users to find relevant information or, in other words, exacerbates the information overload problem. Meanwhile, users in social media can be both passive content consumers and active content producers, causing the quality of user-generated content can vary dramatically from excellence to abuse or spam, which results in a problem of information credibility. Trust, providing evidence about with whom users can trust to share information and from whom users can accept information without additional verification, plays a crucial role in helping online users collect relevant and reliable information. It has been proven to be an effective way to mitigate information overload and credibility problems and has attracted increasing attention.

As the conceptual counterpart of trust, distrust could be as important as trust and its value has been widely recognized by social sciences in the physical world. However, little attention is paid on distrust in social media. Social media differs from the physical world - (1) its data is passively observed, large-scale, incomplete, noisy and embedded with rich heterogeneous sources; and (2) distrust is generally unavailable in social media. These unique properties of social media present novel challenges for computing distrust in social media: (1) passively observed social media data does not provide necessary information social scientists use to understand distrust, how can I understand distrust in social media? (2) distrust is usually invisible in social media, how can I make invisible distrust visible by leveraging unique properties of social media data? and (3) little is known about distrust and its role in social media applications, how can distrust help make difference in social media applications?

The chief objective of this dissertation is to figure out solutions to these challenges via innovative research and novel methods. In particular, computational tasks are designed to {\it understand distrust}, a innovative task, i.e., {\it predicting distrust} is proposed with novel frameworks to make invisible distrust visible, and principled approaches are develop to {\it apply distrust} in social media applications. Since distrust is a special type of negative links, I demonstrate the generalization of properties and algorithms of distrust to negative links, i.e., {\it generalizing findings of distrust}, which greatly expands the boundaries of research of distrust and largely broadens its applications in social media.

Computing Distrust in Social Media

by

Jiliang Tang

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved February 2015 by the
Graduate Supervisory Committee:

Huan Liu, Chair
Guoliang Xue
Jieping Ye
Charu Aggarwal

ARIZONA STATE UNIVERSITY

May 2015

ABSTRACT

A myriad of social media services are emerging in recent years that allow people to communicate and express themselves conveniently and easily. The pervasive use of social media generates massive data at an unprecedented rate. It becomes increasingly difficult for online users to find relevant information or, in other words, exacerbates the information overload problem. Meanwhile, users in social media can be both passive content consumers and active content producers, causing the quality of user-generated content can vary dramatically from excellence to abuse or spam, which results in a problem of information credibility. Trust, providing evidence about with whom users can trust to share information and from whom users can accept information without additional verification, plays a crucial role in helping online users collect relevant and reliable information. It has been proven to be an effective way to mitigate information overload and credibility problems and has attracted increasing attention.

As the conceptual counterpart of trust, distrust could be as important as trust and its value has been widely recognized by social sciences in the physical world. However, little attention is paid on distrust in social media. Social media differs from the physical world - (1) its data is passively observed, large-scale, incomplete, noisy and embedded with rich heterogeneous sources; and (2) distrust is generally unavailable in social media. These unique properties of social media present novel challenges for computing distrust in social media: (1) passively observed social media data does not provide necessary information social scientists use to understand distrust, how can I understand distrust in social media? (2) distrust is usually invisible in social media, how can I make invisible distrust visible by leveraging unique properties of social media data? and (3) little is known about distrust and its role in social media applications, how can distrust help make difference in social media applications?

The chief objective of this dissertation is to figure out solutions to these challenges via innovative research and novel methods. In particular, computational tasks are designed to *understand distrust*, a innovative task, i.e., *predicting distrust* is proposed with novel frameworks to make invisible distrust visible, and principled approaches are develop to *apply distrust* in social media applications. Since distrust is a special type of negative links, I demonstrate the generalization of properties and algorithms of distrust to negative links, i.e., *generalizing findings of distrust*, which greatly expands the boundaries of research of distrust and largely broadens its applications in social media.

DEDICATION

I dedicate my dissertation work to my loving parents, Shuihong Tang and Siyuan Liu, for making me be who I am!

I also dedicate this dissertation to my wife, Hui Liu, for supportng me all the way! Without her help and encouragement, this journey would have not been possible.

ACKNOWLEDGEMENTS

This dissertation is impossible without the help from my advisor Dr. Huan Liu. I would like to thank him to give me large freedom through my Ph.D. to explore various research problems and his excellent advising skills combining patience and guidance make my Ph.D. experience colorful, exciting and productive. I learnt many abilities from him that I can benefit all my life: how to write papers and give presentations, how to find and address challenging problems, and how to establish your career and see the big vision. Dr. Liu is more of a mentor and friend than an advisor for research. He is the mentor because I always seek his suggestions and help for many aspects in my life; and he is also my friend because I feel free to share my personal happiness and sadness with him. Dr.Liu, I cannot thank you enough.

I would like to thank my committee members, Dr. Guoliang Xue, Dr. Jieping Ye, and Dr. Charu Aggarwal, for helpful suggestions and insightful comments. I audited algorithm and optimization courses from Dr. Guoliang Xue, which prepared me with solid technical background and benefited my Ph.D. research a lot. His insightful discussions and commits provide me new angles to rethink about my research. I always consider Dr.Jieping Ye as my secondary advisor because many research ideas were initialized with his discussions. I am impressed by his ability to understand immediately the technical details and the values of my work. Part of this dissertation was done when I was an intern of Dr. Charu Aggarwal. His broad interests and knowledge largely broaden my perspectives on research and greatly expand the boundaries of my research.

I was lucky to work as interns in Yahoo!Labs and IBM Research with amazing colleagues and mentors: Yi Chang, Anlei Dong, Achint Thomas, Dawei Yin, Yandong Liu, Hongbo Deng, and Chikasi Nobata from Yahoo!Labs; Charu Aggarwal, Shu-ping Chang, Fei Wang, Buyue Qian, Guojun Qi, Xiang Wang, Jun Wang and Nan Cao

from IBM Research. Because of you, my life became much easier in new environments; because of you, I enjoyed two wonderful and productive summers; and because of you, I was able to contribute my knowledge to exciting projects. Thank you for everything.

During my Ph.D. study, my friends and colleagues provided me consistent support and encouragement and they deserve a special thank. I am thankful to my colleagues at the Data Mining and Machine Learning Lab: Xufei Wang, Huiji Gao, Xia Hu, Pritam Gundecha, Fred Morstatter , Shamanth Kumar , Ali Abbasi, Reza Zafarani, Rob Trevino, Isaac Jones, Geoffery Barbier, Salem Alelyani, Zhuo Feng, Tahora H. Nazer, Suhang Wang, Suhas Ranganath, Jundong Li, Liang Wu, Ghazaleh Beigi and Kewei Cheng. In particular, thanks to Xufei Wang who helped me write my first paper; thanks to my long-term collaborators Huiji Gao, Xia Hu and Pritam Gundecha and I will remember a lot of deadlines we tried to beat; thanks to Fred Morstatter, Rob Trevino, Shamanth Kumar and Isaac Jones as my English teachers and I will remember any error you corrected and every new word you taught; thanks to Ali Abbasi and Reza Zafarani from whom I learned my presentation skills; thanks to the feature selection team in DMML including Suhang Wang, Jundong Li, Kewei Cheng, Fred Morstatter and Rob Trevino and I believe that our efforts will be paid off. I am also lucky to have you as friends and colleagues in my life: Yuheng Hu, Jiayu Zhou, Yilin Wang, Qiang Zhang, Yingzhou Bi, Guoyong Cai, Yiming Wen, Shijin Li, Ying Wang, Xin Wang and Atish Das Sarma.

Finally, I am deeply indebted to my dear mother and father for their love and strong support during my graduate study. I would like to thank my dear wife Hui Liu for her strong support through all these years to my study. How fortunate I am having her in my life! This dissertation is dedicated to them.

TABLE OF CONTENTS

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

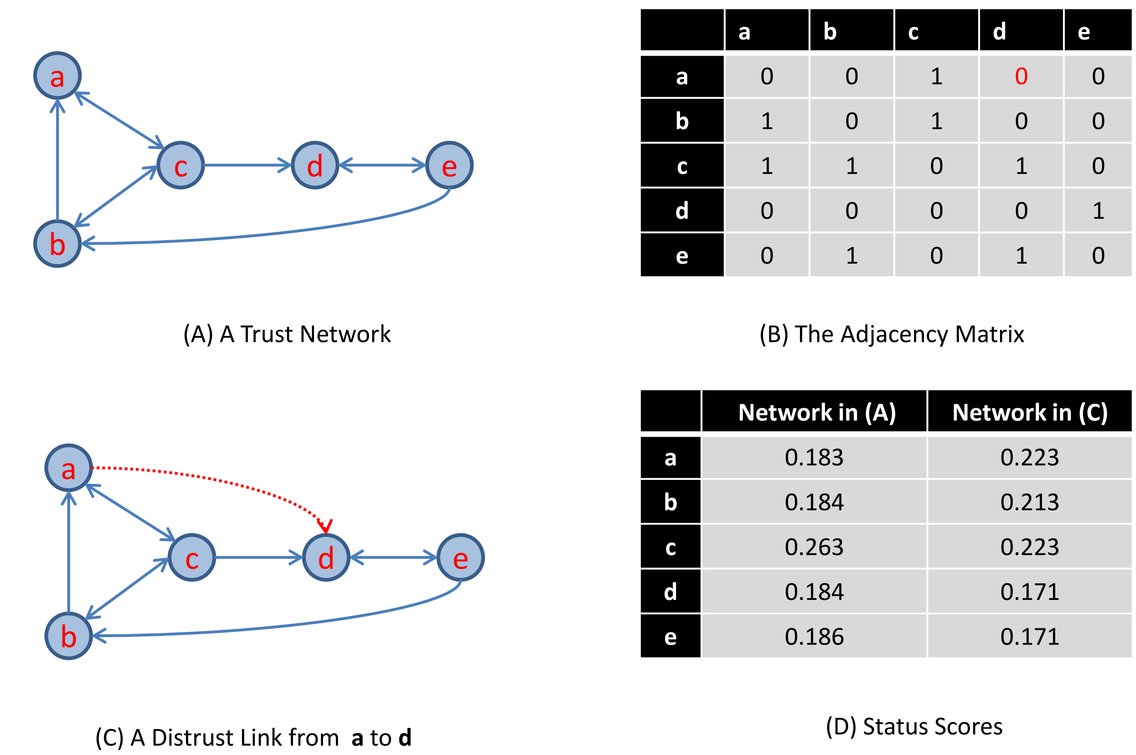
Figure Page

Chapter 1

INTRODUCTION

Social media greatly enables people to participate in online activities and shatters the barrier for online users to create and share information in any place at any time. The pervasive use of social media generates massive data in an unprecedented rate and the information overload problem becomes increasingly severe for social media users. Meanwhile the quality of user-generated content can vary dramatically from excellence content to abuse or spam, resulting in a problem of information credibility [102, 9]. The study and understanding of trust can lead to an effective approach to address both information overload and credibility problems. Trust refers to a relationship between a trustor (the subject that trusts a target entity) and a trustee (the entity that is trusted) [22]. In the context of social media, trust provides evidence about with whom we can trust to share information and from whom we can accept information without additional verification. With trust, we make the mental shortcut by directly seeking information from trustees or trusted entities, which serves a two-fold purpose: without being overwhelmed by excessive information (i.e., mitigated information overload) and with credible information due to the trust placed on the information provider (i.e., increased information credibility). Therefore, trust is crucial in helping social media users collect relevant and reliable information, and computing trust in social media has attracted increasing attention in recent years [87].

Comparing with trust, little attention is paid to distrust in social media. Research with only trust may be biased without considering distrust [23, 83]. Typically, trust relations can be represented as a trust network and the adjacency matrix is adopted to represent the trust network where “0” is used to indicate no trust as illustrated



(A) A Trust Network

	a	b	c	d	e
a	0	0	1	0	0
b	1	0	1	0	0
c	1	1	0	1	0
d	0	0	0	0	1
e	0	1	0	1	0

(B) The Adjacency Matrix

(C) A Distrust Link from **a** to **d**

	Network in (A)	Network in (C)
a	0.183	0.223
b	0.184	0.213
c	0.263	0.223
d	0.184	0.171
e	0.186	0.171

(D) Status Scores

Figure 1.1: The Impact of Distrust.

in Figure 1.1 (A) and (B). However, this representation may not be representative since a zero score cannot distinguish between distrust and no trust. For example, a distrust relation may exist from node a to node d as shown in Figure 1.1 (C). Furthermore, ignoring distrust in online applications may lead to over-estimation of the effects of trust [93]. The first column of Table (D) in Figure 1.1 shows reputation scores, calculated using PageRank [61], of nodes in the network of Figure 1.1 (A); while the second column shows reputation scores, calculated by a variant of Pagerank taking into account distrust [93], of nodes in the network of Figure 1.1 (C). The only difference between networks in Figure 1.1 (A) and (C) is a distrust relation from a to d in the network (C); clearly, the small difference significantly affects the statuses of the nodes. These findings suggest that distrust could be as important as trust.

1.1 Research Challenges

Social scientists have studied distrust in the physical world in reducing uncertainty and vulnerability associated with critical decisions [63, 3]. Social media differs from the physical world and computing distrust in social media face challenges:

- Social scientists understand distrust from the perspectives of formation mechanisms and constructs [40, 12]. One understanding is that distrust is the negation of trust while an alternative understanding is that distrust has added value over trust. There is still no consensus about the understanding of distrust in social sciences [63, 3, 29]. However, understanding distrust with social media data is inherently difficult. Social media data is based on passive observations with a large number of online users and lacks of necessary information social scientists use to study distrust where interactions with users are required. Since methods from social sciences are not directly applicable, the first challenge is how we can *understand* distrust in social media.

- It is suggested in research [37, 24] that trust is a desired property while distrust is an unwanted one for an online social community. Therefore, various online services such as Ciao[1], eBay[2] and Epinions[3] implement trust mechanisms to help users to better use their services, but few of them allow online users to specify distrust relations. Since distrust is usually unavailable in social media, the second challenge is how we can make *unavailable* distrust available in social media?

- An ultimate assessment of the utility of distrust is its impact on real-world

[1] http://www.ciao.co.uk/

[2] http://www.ebay.com/

[3] http://www.epinions.com/

applications. We observe successful applications of trust such as node classification [65, 19] and social recommendation [82]. Social theories such as homophily [56] and social influence [50] serve as the groundwork in many trust applications; but they may not applicable to distrust [83]. Since we know too little about distrust and applying distrust may not be carried out by simply extending these of trust, the third challenge is how we can *apply* distrust in social media applications?

1.2 Contributions

The aforementioned challenges present a series of interesting research questions - (1) is distrust the negation of trust? and does distrust has added value over trust? (2) can invisible distrust be discovered? (3) how can distrust help make difference in social media applications? and (4) is research of distrust generalizable to negative links in social media? One of the chief objectives of this dissertation is to figure out answers to these questions via innovative research. The contributions of this dissertation are summarized as:

- The unique properties of social media determines that innovative methods should be developed in order to *understand distrust*. We design two computational tasks by leveraging data mining and machine learning techniques to enable the computational understanding of distrust in social media. The first task is to predict distrust from only trust, which is designed to seek an answer for the question of "is distrust the negation of trust?"; and the second task is to predict trust with information from distrust, which is designed for the question of "does distrust have added value over trust?"

- We propose a new research task, i.e., *predicting distrust*, which aims to automatically predict distrust when distrust is unavailable by leveraging available sources in social media. We make a number of important findings about distrust and develop an unsupervised framework dTrust and a supervised framework NeLP, which can predict distrust accurately by using trust and content-centric user interactions.

- We provide principled approaches to exploit distrust for social media applications, i.e., *applying distrust*. In detail, we use node classification and recommendation as two representative applications to illustrate the importance and the added value of distrust in social media applications.

- We generalize research about distrust to negative links, i.e., *generalizing findings of distrust*. In particular, we find that properties and algorithms of distrust can be generalized to negative links, which greatly expands the boundaries of the research of computing distrust and broadens its applications.

1.3 Organization

The remainder of this dissertation is organized as follows. In Chapter 2, we introduce some basic concepts, our research on computing trust in social media, and background of distrust in social science. In Chapter 3, we first introduce possible representations of distrust, then investigate properties of distrust and detail two computational tasks to seek answers for the questions of "is distrust the negation of trust?" and "does distrust have added value over trust?". In Chapter 4, we formally define the problem of distrust prediction, perform analysis on distrust and introduce the details about the unsupervised framework dTrust and the supervised framework NeLP. In Chapter 5, we study two representative applications of distrust with a framework NCSSN for node classification and a framework RecSSN for recommendation. In

Chapter 6, we investigate the generalization of properties of distrust to negative links, expand dTrust and NeLP to predict negative links and apply application frameworks NCSSN and RecSSN for negative links. We conclude the dissertation and point out broader impacts and promising research directions in Chapter 7.

Chapter 2

FOUNDATIONS AND PRELIMINARIES

Before studying distrust in social media, we have prepared ourselves with research on computing trust in social media. In this section, we will briefly introduce our research on computing trust in social media first, and then give background about distrust research in social sciences.

2.1 Computing Trust in Social Media

There are three major computational tasks for trust - representing trust, measuring trust and applying trust [87, 89]. Below we introduce the research we have done for computing trust in social media.

Representing Trust: It aims to represent trust relations among users and trust representations can be roughly divided into probabilistic representations and gradual representations [94]. Traditional trust representation considers trust as a single concept, however, trust is a complex concept with multiple dimensions. Therefore, we propose two multi-dimensional trust representations - mTrust [78] and eTrust [79, 75]. Trust is context dependent and trusting someone on one topic does not necessarily mean he will be trusted on others, hence, mTrust is proposed to capture multi-faceted trust, i.e., trust relations under different contexts. As humans interact, trust evolves and eTrust is proposed to capture trust evolution.

Measuring Trust: It measures how much a certain user can be trusted by other users from the community. From different perspectives, trust metrics can be classified differently [87]. From a personalization perspective, trust metrics can be classified as global [30] and local trust metrics [21]. From a methodology perspective, trust metrics can be supervised [59] or unsupervised [108]. From a network perspective, trust metrics can be binary or continuous [87]. A few factors can influence people to establish trust relations and a user usually establishes trust relations with a small proportion of users in the trust network, resulting in the adjacent matrix very sparse and low-rank, hence users can have a more compact but accurate representation in a low-rank space. hTrust is an unsupervised trust measurement framework based on low-rank matrix factorization [77]. Online trust relations follow a power law distribution, suggesting that a small number of users specify many trust relations while a large proportion of users specify a few trust relations. The power law distribution indicates that the available trust relations may not be enough to guarantee the success of existing trust measurements. On the other hand, there are many theories developed to explain the formation of trust such as homophily [56] and status theory [39], and these social theories may be helpful to mitigate the data sparsity problem. We propose hTrust [77] and sTrust [97] to exploit homophily effect and status theory to improve trust measurements for users with few or no trust relations. User preferences may evolve over time and we also consider temporal dynamics in trust metrics [7].

Applying Trust: It aims to incorporate trust to help social media applications. Recommendation is one of the most popular and important applications of trust [53, 82, 2]. Existing trust-aware recommendation systems can be divided into memory-based methods [21, 27] and model-based methods [46, 48]. In the physical world, people seek recommendations from their trusted friends and they are also likely to accept recommendations from trustworthy users. Therefore we proposed LOCABEL to exploit local and global trust for recommendation [81]. Users may think reviews from their trusted users more useful and CAP is proposed to apply trust in review recommendation in online review websites such as eBay [76]. An comprehensive overview about trust-aware recommender systems can be found in our survey paper in [82].

2.2 Distrust in Social Sciences

In social sciences, the conceptual counterpart of trust, distrust, is considered as important and complex as trust [54, 37, 24, 13]. For example, [68, 12] claim that trust and distrust help a decision maker reduce uncertainty and vulnerability (i.e., risk) associated with decision consequences; and [13] indicates that only distrust can irrevocably exclude services from being selected at all. There is an enduring problem about distrust - what is the relation between trust and distrust. Answering this question has its significance. If trust and distrust are the same, lack of distrust research matters little, however, if they are different, the lack of distrust research could be problematic because distrust may have unique impact. Some researchers believe distrust simply means a low level of trust, hence evidence of high trust was always regarded as being that of low distrust, and outcomes of high trust would be identical to those of low distrust [63, 3, 29]. Others believe distrust is a concept entirely separate from trust [40, 36]. Therefore distrust and trust can coexist, and they have different antecedents and consequents [55]. For example, in [40], three reasons are proposed to prove that trust and distrust are separate - (1) they separate empirically; (2) they coexist; and (3) they have different antecedents and consequents. There is still no consensus answer about this problem, and some social scientists consider distrust as the “darker” side of trust [51].

Chapter 3

UNDERSTANDING DISTRUST IN SOCIAL MEDIA

Social scientists understand distrust from the perspectives of formation mechanisms and constructs [40, 12] and have recognized that distrust helps a decision maker reduce the uncertainty and vulnerability (i.e., risk) associated with decision consequences [12], and in some cases, plays a more critical role than trust [68, 54]. Large-scale social media data does not contain necessary information social scientists ascribe in their studying distrust. In social media, distrust is embedded with rich but passively observed user-generated content and interactions. Understanding distrust in social media requires new methods since those from social sciences are not directly applicable, but can be helpful in our search of new methods. More specifically, we aim to answer the following research questions: (1) what are the properties of distrust? (2) is distrust the negation of trust? and (3) does distrust have added value over trust?

3.1 Representing Distrust

Computational models for trust depends on certain trust representations [87]; hence an immediate question for distrust is how to represent distrust. We propose to represent distrust with trust, which can not only avoid possible biases because of the ignorance of trust in distrust representations but also help us better understand the role and added value of distrust over trust in applications.

Distrust representations are substantially different with different understandings about distrust over trust. If distrust is considered as the negation of trust [63, 3], high (or low) trust would be identical to those of low (or high) distrust [29]. In this

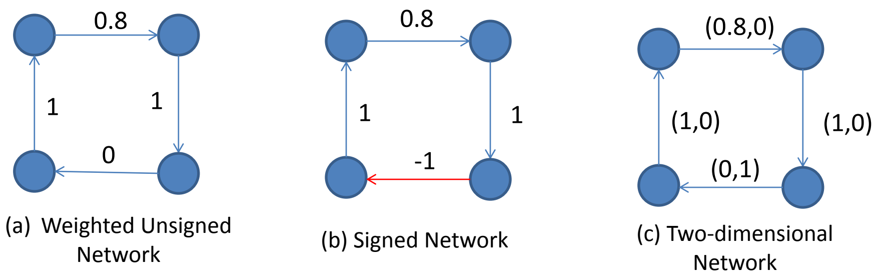


Figure 3.1: Network Understandings of Representations of Trust and Distrust.

case, we represent trust and distrust are two ends of the same conceptual spectrum. If distrust is not the negation of trust [40, 36], there are two views about the relations between trust and distrust as:

- Trust and distrust are viewed as tightly related features in a single structure [38]. Hence we add positive and negative signs to represent trust and distrust respectively, and we keep the semantics of a zero score in the representation; and
- Distrust is viewed as a distinct dimension from trust about users [69]. Hence we add a new dimension about users to represent distrust.

To further understand aforementioned three representations better, we show these representations from the network perspective as demonstrated in Figure 3.1. When we consider distrust as low trust, the trust and distrust network is a weighted unsigned network as shown in Figure 3.1(a); when we add signs to represent trust and distrust, the resulting network is a weighted signed network as shown in Figure 3.1(b); while we add a new dimension to represent distrust, the trust and distrust network is a weighted multi-dimensional unsigned network as shown in Figure 3.1(c).

Let $\mathcal{U} = \{u_1, u_2, \ldots, u_n\}$ be the sets of users where n is the number of users. If we consider trust and distrust links as tightly related features in a single network. We represent both trust and distrust links into one adjacency matrix $\mathbf{F} \in \mathbb{R}^{n \times n}$ where

$\mathbf{F}_{ij} = 1$, $\mathbf{F}_{ij} = -1$ and $\mathbf{F}_{ij} = 0$ denote trust, distrust and missing links from u_i to u_j, respectively. If we tread distrust as a distinct dimension from trust about users, we use two adjacency matrices to represent trust and distrust links, respectively. In particular, it uses $\mathbf{T} \in \mathbb{R}^{n \times n}$ to represent trust links where $\mathbf{T}_{ij} = 1$ and $\mathbf{T}_{ij} = 0$ denote a trust link and a missing link from u_i to u_j. Similarly $\mathbf{D} \in \mathbb{R}^{n \times n}$ is used to represent distrust links where $\mathbf{D}_{ij} = 1$ and $\mathbf{D}_{ij} = 0$ denote a distrust link and a missing link from u_i to u_j.

It is easy to convert one representation into the other representation with the following rules: $\mathbf{F} = \mathbf{T} - \mathbf{D}$, and $\mathbf{T} = \frac{|\mathbf{F}| + \mathbf{F}}{2}$ and $\mathbf{D} = \frac{|\mathbf{F}| - \mathbf{F}}{2}$ where $|\mathbf{F}|$ is the absolution of $\mathbf{F}$.

3.2 Properties of Distrust

Properties of trust such as transitivity, asymmetry and homophily help determine the value of trust[20, 77]. Hence, to understand distrust, it is natural to start with exploring properties of distrust. Naturally, we question if there are some obvious connections between these properties of trust to distrust. The proposed research can help us understand how unique distrust is and the intrinsic differences between trust and distrust.

Before investigations, we first introduce the dataset we used. Trust mechanisms are implemented by various online services; however, few of them allow users to establish distrust relations. Although the product review site Epinions allows users to trust and distrust other users, distrust relations are unavailable to the public. For the research purpose, a dataset with distrust relations was given by Epinions staff. We preprocess the data by filtering users without any trust and distrust relations. This dataset includes trust and distrust relations, user-item ratings, user-review authorship relations and user-review helpfulness ratings. The statistics of the dataset are shown

# of Users	30,455
# of Trust Relations	363,773
# of Distrust Relations	46,196
# of Users Receiving Distrust	9,513
# of Users Creating Distrust	5,324
# of Items	89,270
# of Ratings	562,355
# of Reviews	1,197,816
# of Helpfulness Ratings	10,341,893
Avg of Helpfulness Rating Score	4.7129
Avg of Rating Score	3.9053

Table 3.1: Statistics of the Epinions.

in Table 3.1.

3.2.1 Transitivity

Transitivity is a primary property of trust and it describes that trust can be passed between people [29, 20]. For example, if user u_i trusts user u_j, and user u_j trusts user u_k, then transitivity indicates that with a high probability, user u_i will trust user u_k. Here we study the property of distrust with respect to transitivity. Note that we use x+y, x-y, and x?y to denote the observations of a trust, a distrust and a missing relation from user x to user y, respectively.

To investigate the transitivity property of distrust, we first find all pairs of relations $\langle u_i\text{-}u_j,u_j\text{-}u_k\rangle$, and check whether u_i and u_k are with a trust (u_i+u_k), or a distrust (u_i-u_k), or a missing relation (u_i?u_k). We conduct a similar process for trust, and the results are demonstrated in Table 3.2. For the first calculation, we consider all

$\langle u_i$,$u_k\rangle$ pairs (i.e., u_i+u_k, u_i-u_k, or u_i?u_k) and use "P1" to denote the percentage of pairs of $\langle$ u_i,$u_k\rangle$ with a trust, a distrust or a missing relation over all pairs. For the second calculation, we only consider$\langle u_i$,$u_k\rangle$ pairs with observed relations (i.e., trust u_i+u_k or distrust u_i-u_k), and adopt "P2" to represent the percentage of $\langle$ u_i,$u_k\rangle$ with a trust or a distrust relation over pairs with observed relations (i.e., u_i+u_k or u_i-u_k).

Golbeck suggests that trust is not perfectly transitive in the mathematical sense and is conditionally transitive [20], which is especially true for social media data since users in social media are world-widely distributed and there are many pairs of user not knowing each other. A trust relation u_i+u_k only takes 11.46% (P1) of all pairs of $\langle u_i$,u_k $\rangle$ for trust. However, among pairs with observed relations, u_i+u_k takes as high as 97.75% (P2), which suggests the transitivity property for trust - if u_i establishes a relation with u_k, it is likely to be a trust relation. For distrust, the percentages of u_i-u_k and u_i+u_k are comparable. u_i-u_k suggests transitivity, which can be explained by status theory; while u_i+u_k can be explained by balance theory [8, 25] as "the enemy of your enemy is your friend."

3.2.2 Asymmetry

The asymmetry of trust is also important and suggests that for two people involved in a relation, trust is not necessarily identical in both directions [22]. For example, if u_i trusts u_j, one cannot infer that u_j trusts u_i. Next we examine the property of distrust in term of asymmetry.

For each trust relation u_i+u_j (or each distrust relation u_i-u_j), we check the possible relations from u_j to u_i, and the results are illustrated in Table 3.3. Note that in Table 3.3 the numbers in parentheses are the percentages of the corresponding relations over all possible relations. We observe 37.71% mutual trust relations, but only 5.86% mutual distrust relations. These results suggest that trust is asymmetric,

Trust			
Types	Number	P1	P2
$\langle u_i+u_j,u_j+u_k\rangle$, $u_i?u_k$	25,584,525	88.34%	N.A
$\langle u_i+u_j,u_j+u_k\rangle$, u_i+u_k	3,320,991	11.46%	97.75%
$\langle u_i+u_j,u_j+u_k\rangle$, u_i-u_k	76,613	0.2%	2.25%

Distrust			
Types	Number	P1	P2
$\langle u_i$-u_j,u_j-$u_k\rangle$, $u_i?u_k$	716,340	91.70%	N.A
$\langle u_i$-u_j,u_j-$u_k\rangle$, u_i+u_k	38,729	4.96%	59.73%
$\langle u_i$-u_j,u_j-$u_k\rangle$, u_i-u_k	26,114	3.34%	40.27%

Table 3.2: Transitivity of Trust and Distrust.

	u_j+u_i(%)	u_j-u_i(%)	$u_j?u_i$(%)
u_i+u_j	136,806(37.61)	967(0.27)	226,000(62.13)
u_i-u_j	967(2.09)	2,623(5.86)	42,606(92.23)

Table 3.3: Asymmetry of Trust and Distrust.

and distrust is even more asymmetric.

3.2.3 Similarity

Ziegler et al. [107] point out that there is a strong and significant correlation between trust and similarity and users with trust relations are more similar than those without. Next we investigate the relations between distrust and similarity. In Epinions, users can give a score from 1 to 5 to rate various items and we define user similarities as their rating similarities [77]. We use the following three measures to compute the similarity between u_i and u_j as:

	CI	COSINE	CI-COSINE
Distrust ($\mathbf{s}_d$)	0.4994	0.0105	0.0142
Trust ($\mathbf{s}_t$)	0.6792	0.0157	0.0166
Random Pairs ($\mathbf{s}_r$)	0.1247	0.0027	0.0032

Table 3.4: Similarity for Trust and Distrust.

- CI: the number of common items rated by both u_i and u_j;
- COSINE: the cosine similarity between the rating scores of u_i and u_j over all items; and
- CI-COSINE: the cosine similarity between the rating scores of u_i and u_j over their commonly rated items.

We calculate three similarities for each distrust relation, i.e., distrust similarity d_s, trust similarity t_s, and random similarity r_s. For example, for the distrust relation u_i-u_j, indicating that u_i distrusts u_j, d_s is the rating similarity between u_i and u_j, t_s the similarity between u_i and a randomly chosen user who has a trust relation with u_i, and r_s the similarity between u_i and a randomly chosen user without a distrust relation with u_i. Over all distrust relations, finally we obtain three similarity vectors, $\mathbf{s}_d$, $\mathbf{s}_t$, and $\mathbf{s}_r$. $\mathbf{s}_d$ is the set of all distrust similarities d_s, $\mathbf{s}_t$ the set of t_s, and $\mathbf{s}_r$ the set of r_s. The means of $\mathbf{s}_d$, $\mathbf{s}_t$ and $\mathbf{s}_r$ are shown in Table 3.4, respectively. We observe that the means of distrust are larger than those of random but smaller than those of trust over all three similarity measures.

To see the significance, we conduct a series of two-sample t-test on $\{\mathbf{s}_d, \mathbf{s}_r\}$, $\{\mathbf{s}_t, \mathbf{s}_r\}$, and $\{\mathbf{s}_t, \mathbf{s}_d\}$. For two vectors $\{\mathbf{x}, \mathbf{y}\}$, the null hypothesis H_0, and the alternative hypothesis H_1 are defined as:

$$H_0 : \mathbf{x} <= \mathbf{y} \qquad H_1 : \mathbf{x} > \mathbf{y}. \tag{3.1}$$

	CI	COSINE	CI-COSINE
$H_0 : \mathbf{s}_d <= \mathbf{s}_r$ $H_1 : \mathbf{s}_d > \mathbf{s}_r$	9.57e-87	1.19e-120	4.88e-45
$H_0 : \mathbf{s}_t <= \mathbf{s}_r$ $H_1 : \mathbf{s}_t > \mathbf{s}_r$	1.71e-132	5.83e-157	3.72e-108
$H_0 : \mathbf{s}_t <= \mathbf{s}_d$ $H_1 : \mathbf{s}_t > \mathbf{s}_d$	7.84e-23	1.99e-19	9.32e-17

Table 3.5: P-values for t-test Results.

The null hypothesis is rejected at significance level $\alpha = 0.01$ with p-values shown in Table 3.5. Means in Table 3.4 and evidence from t-test suggest that - (1) users with distrust are likely to be more similar than those without; (2) users with trust are likely to be more similar than those without; and (3) users with trust are likely to be more similar than those with distrust.

3.2.4 Discussion

Aforementioned empirical investigations indicate that distrust is not transitive, highly asymmetric and neither similarity nor dissimilarity. Through this comparative study on properties of trust and distrust, we can conclude that (1) the properties of trust cannot be extended to distrust; and (2) distrust presents distinct properties.

3.3 Constructing Distrust from Only Trust

Some social scientists believe distrust as the negation of trust - trust and distrust are two ends of the same conceptual spectrum, and distrust can be suggested by low trust [63, 3]. To seek an answer to the question of "is distrust the negation of trust?", we design the task of constructing distrust from only trust. The reasoning behind this task is if distrust is the negation of trust, distrust can be suggested for pairs of users with low trust and we can accurately construct distrust from only trust. Therefore *the problem of constructing distrust from only trust boils down to the problem of predicting* *low trust with trust.* Trust scores of pairs of users in the same trust network can be computed via existing trust prediction algorithms.

The general framework for the task is shown in Algorithm 1. The input of the framework includes trust $\mathbf{T}$ and a trust predictor f. For a pair of users without trust $\langle u_i, u_j \rangle$, we use f to predict a trust score $\tilde{\mathbf{T}}_{ij}$ from u_i to u_j and then suggest pairs with low trust scores as distrust. We choose two representative trust predictors - trust propagation [23] and the one we proposed in [77] based on matrix factorization.

Algorithm 1 The framework of *Task 1* to predict distrust from only trust

```
Input: User-user trust relation matrix $\mathbf{T}$, and a trust predictor $f$
Output: Ranking list of pairs of users
for Each pair of users without trust $\langle u_i, u_j \rangle$ do
    Predicting the trust score of $\tilde{\mathbf{T}}_{ij}$ from $u_i$ to $u_j$ by $f$
end for
Ranking pairs of users (e.g., $\langle u_i, u_j \rangle$) according to $\tilde{\mathbf{T}}_{ij}$ in an ascending order.
```

3.3.1 Trust Prediction based on Matrix Factorization

Trust has some well-known properties such as transitivity, asymmetry and correlation with user preference similarity, which lay the groundwork for meaningful and effective trust prediction models. Let $\mathbf{U}_i \in \mathbb{R}^d$ denote the user preference vector of u_i. In [77], we propose to model a trust relation from u_i to u_j as $\mathbf{T}_{ij} = \mathbf{U}_i \mathbf{V} \mathbf{U}_j^\top$ where $\mathbf{V} \in \mathbb{R}^{d \times d}$ captures the correlations among use preferences. We can verify that the proposed model can capture several important properties of trust such as transitivity, asymmetry and similarity. For example, $\mathbf{V}$ could be asymmetric, therefore $\mathbf{T}_{ij} = \mathbf{U}_i \mathbf{V} \mathbf{U}_j^\top$ could be unequal to $\mathbf{T}_{ji} = \mathbf{U}_j \mathbf{V} \mathbf{U}_i^\top$, which captures the property of asymmetry. Let $\mathbf{U} = \{\mathbf{U}_1, \mathbf{U}_2, \dots, \mathbf{U}_n\}$ be the user preference matrix. $\mathbf{U}$ and $\mathbf{V}$ can be obtained via solving the following low-rank matrix factorization problem:

$$\min_{\mathbf{U},\mathbf{V}} \quad \|\mathbf{T} - \mathbf{U}\mathbf{V}\mathbf{U}^\top\|_F^2 + \alpha\|\mathbf{U}\|_F^2 + \beta\|\mathbf{V}\|_F^2 \tag{3.2}$$

where terms of $\alpha\|\mathbf{U}\|_F^2 + \beta\|\mathbf{V}\|_F^2$ are introduced to avoid overfitting. With the learned $\mathbf{U}$ and $\mathbf{V}$, the estimated user-user trust relation matrix $\tilde{\mathbf{T}}$ is obtained as $\tilde{\mathbf{T}} = \mathbf{U}\mathbf{V}\mathbf{U}^\top$.

3.3.2 Evaluation

In this subsection, we conduct experiments to answer the following question: is distrust the negation of trust? To answer the question, we check how accurately we can predict distrust from only trust.

Experimental Settings

We first introduce the experimental setting for this evaluation. $\mathcal{A}_T = \{\langle u_i, u_j\rangle | \mathbf{T}_{ij} = 1\}$ is the set of pairs of users with trust relations, and $\mathcal{A}_D = \{\langle u_i, u_j\rangle | \mathbf{D}_{ij} = 1\}$ is the set of pairs of users with distrust relations. The pairs in both $\mathcal{A}_T$ and $\mathcal{A}_D$ are sorted in chronological order in terms of the time when they established relations. We assume that until time t, $x\%$ of pairs in $\mathcal{A}_T$ establish trust relations, denoted as $\mathcal{A}_T^x$, and we use $\mathcal{A}_D^x$ to denote pairs of users in $\mathcal{A}_D$ establishing distrust until time t. x is varied as $\{50, 55, 60, 65, 70, 80, 90, 100\}$. For each x, we use $\mathcal{A}_T^x$ to predict $\mathcal{A}_D^x$ from N_T^x where N_T^x is the negation of $\mathcal{A}_T^x$ as shown in Figure 3.2.

We follow a common metric for trust evaluation in [41, 77] to assess the prediction performance. In detail, each predictor ranks pairs in N_T^x in **ascending** order of confidence and we take the first $|\mathcal{A}_D^x|$ pairs as the set of predicted distrust relations, denoting $\mathcal{A}_D^p$. Then the performance is computed as,

$$M_1 = \frac{|\mathcal{A}_D^x \cap \mathcal{A}_D^p|}{|\mathcal{A}_D^x|} \tag{3.3}$$

where $|\cdot|$ denotes the size of a set.

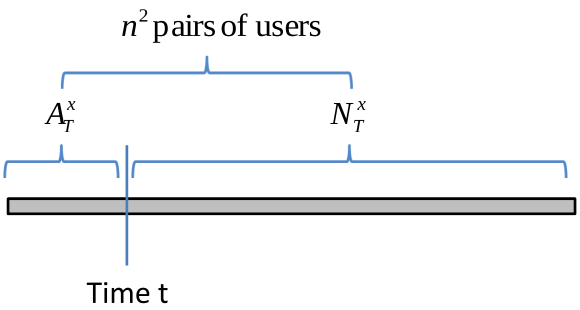


Figure 3.2: Experimental Settings for Constructing Distrust from Only Trust.

Experimental Results

The results are shown in the Table 3.6. “dTP”, “dMF”, and “dTP-MF” and “Random” in the table are defined as follows:

- *dTP:* this distrust predictor obtains trust scores for pairs of users based on trust propagation, and then suggests distrust relations for pairs with low trust scores;
- *dMF:* this distrust predictor computes trust scores for pairs of users based on matrix fatorization, and then predict pairs with low trust scores as distrust relations;
- *dTP-MF:* this distrust predictor combines results of dTP and dMF to infer distrust relations; and
- *Random:* this distrust predictor randomly guesses pairs of users with distrust relations.

If distrust is the negation of trust, low trust scores should accurately indicate distrust. However, we observe that the performance of dTP, dMF and dTP-MF is consistently worse than that of the randomly guessing (Random). These results suggest that low trust scores cannot be used to predict distrust; hence distrust is not

x (%)	dTP ($\times10^{-5}$)	dMF($\times10^{-5}$)	dTP-MF($\times10^{-5}$)	Random($\times10^{-5}$)
50	4.8941	4.8941	4.8941	5.6824
55	5.6236	5.6236	5.6236	8.1182
60	7.1885	7.1885	7.1885	15.814
65	11.985	11.985	11.985	19.717
70	13.532	13.532	13.532	18.826
80	10.844	10.844	10.844	16.266
90	12.720	12.720	12.720	25.457
100	14.237	14.237	14.237	29.904

Table 3.6: Performance Comparison of Predicting Distrust from Only Trust.

the negation of trust. Social scientists, who support distrust not the negation of trust, argue that pairs of users with untrust can have very low trust scores [51], which is especially correct for users in social media since they are world-widely distributed.

3.4 Trust Prediction with Information from Distrust

An alternative understanding is that distrust is not the negation of trust and it has added value over trust [40, 36]. To seek an answer to the question of "does distrust have added value over trust?", we design the task of trust prediction with information from distrust. The intuition behind this task is if distrust has added value over trust, distrust should provide extra information about users and we shall be able to predict trust better with distrust. An illustration of the problem of trust prediction with information from distrust is shown in Figure 3.3 - the input of traditional trust prediction is only old trust relations; we propose to also use distrust information as shown in the dashed box in Figure 3.3. In [23], two strategies are investigate to incorporate distrust into trust propagation: (1) one step distrust propagation -

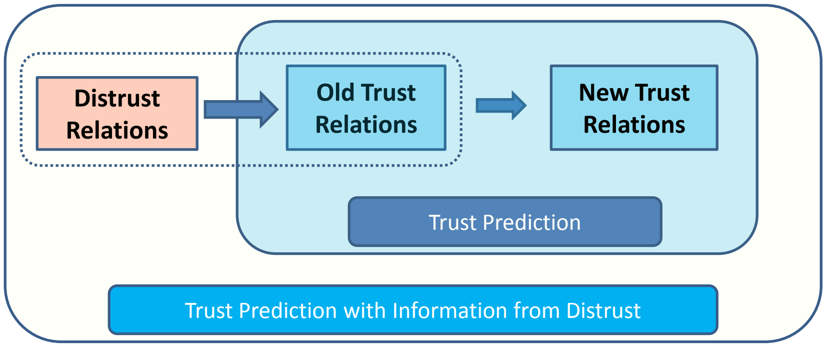


Figure 3.3: Trust Prediction with Information from Distrust.

first propagating trust for multiple steps and then propagating one-step distrust; and (2) multiple step distrust propagation - propagating trust and distrust propagate together for multiple steps. Next we will investigate how to incorporate distrust into the matrix factorization based trust prediction [77].

3.4.1 Matrix Factorization based Trust Prediction with Distrust Information

We view trust and distrust as tightly related features in a single structure (or a signed network) and choose *adding signs* to represent trust and distrust as $\mathbf{F} = \mathbf{T} - \mathbf{D}$. The advantages of this representation are two-fold. First, we can extend the matrix factorization based trust prediction to incorporate distrust information as $\mathbf{F}_{ij} = \mathbf{U}_i\mathbf{V}\mathbf{U}_j^\top$. Second, by adding signs, trust and distrust relations are represented as a signed network and social theories for signed networks such as balance theory can be used to facilitate the task. Below we introduce how to model balance theory.

We use s_{ij} to denote the sign of the relation between u_i and u_j where $s_{ij} = 1$ (or $s_{ij} = -1$) if we observe a trust relation (or a distrust relation) between u_i and u_j. With these notations, balance theory suggests that a triad $\langle u_i, u_j, u_k \rangle$ is balanced if - (1) $s_{ij} = 1$ and $s_{jk} = 1$, then $s_{ik} = 1$; or (2) $s_{ij} = -1$ and $s_{jk} = -1$, then $s_{ik} = 1$. For a triad $\langle u_i, u_j, u_k \rangle$, there are four possible sign combinations **A**(+,+,+), **B**(+,+,-

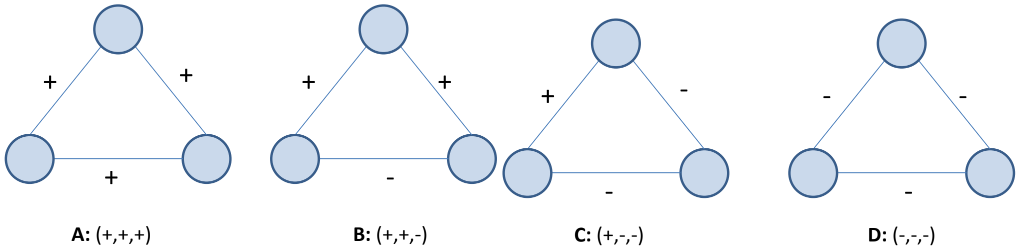


Figure 3.4: An Illustration of Balance Theory.

) **C**(+,-,-) and **D**(-,-,-). According to balance theory, only **A**(+,+,+) and **C**(+,-,-) are balanced. We examine all triads in the studied dataset and find that more than 90% of them are balanced. This result suggests that balance theory is a principle to understand the formation of trust and distrust relations.

For each user u_i, we introduce a one-dimensional latent factor r_i to model balance theory, and we further assume that the relation between u_i and u_j due to the effect of balance theory is captured as $\mathbf{F}_{ij} = r_i r_j$. To show that $\mathbf{F}_{ij} = r_i r_j$ can capture balance theory, we need to prove that - (1) *Case 1:* if $sign(\mathbf{F}_{ij}) = 1$ and $sign(\mathbf{F}_{jk}) = 1$, we can prove that $sign(\mathbf{F}_{ik}) = 1$; and (2) *Case 2:* if $sign(\mathbf{F}_{ij}) = -1$ and $sign(\mathbf{F}_{jk}) = -1$, we can prove that $sign(\mathbf{F}_{ik}) = 1$.

Let us first prove *Case 1.* If $sign(\mathbf{F}_{ij}) = 1$ and $sign(\mathbf{F}_{jk}) = 1$, we have $sign(r_i r_j) = 1$ and $sign(r_j r_k) = 1$; by multiplying $sign(r_i r_j)$ and $sign(r_j r_k)$, we have $sign(r_i r_j r_j r_k) = 1$. Since $sign(r_j^2) = 1$, we get $sign(r_i r_k) = 1$, i.e., $sign(\mathbf{F}_{ik}) = 1$. We can use a similar process to prove *Case 2.* If $sign(\mathbf{F}_{ij}) = -1$ and $sign(\mathbf{F}_{jk}) = -1$, we have $sign(r_i r_j) = -1$ and $sign(r_j r_k) = -1$; by multiplying $sign(r_i r_j)$ and $sign(r_j r_k)$, we have $sign(r_i r_j r_j r_k) = 1$. Since $sign(r_j^2) = 1$, we get $sign(r_i r_k) = 1$, i.e., $sign(\mathbf{F}_{ik}) = 1$.

The proposed framework disMF for the problem of trust prediction with distrust information is to solve the following optimization problem,

$$\min_{\mathbf{U},\mathbf{V},\mathbf{r}} \|\mathbf{F} - \mathbf{U}\mathbf{V}\mathbf{U}^\top - \lambda \mathbf{r}\mathbf{r}^\top\|_F^2 + \alpha\|\mathbf{U}\|_F^2 + \beta\|\mathbf{V}\|_F^2 + \eta\|\mathbf{r}\|_2^2, \tag{3.4}$$

where $\mathbf{r} = [r_1, r_2, \ldots, r_n]^\top$; the term $\|\mathbf{r}\|_2^2$ is introduced to avoid overfitting; and the parameter λ is used to control the contributions from balance theory. A local minimum of Eq. (4.9) can be obtained through a gradient decent optimization method.

3.4.2 Evaluation

In this subsection, we conduct experiments to answer the following question: does distrust have added value in trust prediction? To answer the question, we examine whether the performance of trust prediction is improved by exploiting distrust.

Experimental Settings

Before going to the detailed evaluation, we first introduce the experimental setting. We use $\mathcal{O} = \{\langle u_i, u_j\rangle | \mathbf{T}_{ij} \neq 1\}$ to denote the set of pairs of users without trust relations. Assume that at time t, $x\%$ of $\mathcal{A}_T$ have estabilshed trust relations. We choose these $x\%$ as old trust relations $\mathcal{A}_T^x$, and the remaining $1 - x\%$ as new trust relations $\mathcal{A}_T^n$ to predict. We use $\mathcal{A}_D^x$ to denote the subset of pairs in $\mathcal{A}_D$, which have established distrust before t. We vary x as $\{50, 55, 60, 65, 70, 80, 90\}$. For each x, we also repeat the experiments 10 times and report the average performance. The experimental setting is illustrated in Figure 3.5 where N_T^x denotes the set of pairs without trust relations at time t.

For each x, we use old trust relations $\mathcal{A}_T^x$ and distrust relations $\mathcal{A}_D^x$ to predict new trust relations $\mathcal{A}_T^n$. Each predictor ranks pairs in N_T^x in **decreasing** order of confidence and we take the first $|\mathcal{A}_T^n|$ pairs as the set of predicted trust relations,

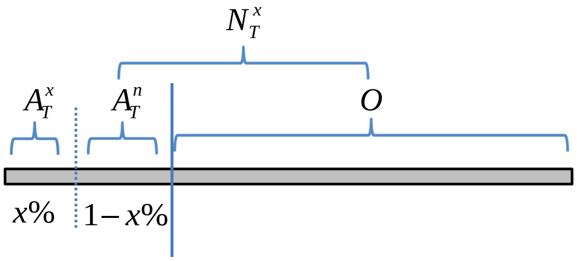


Figure 3.5: Experimental Setting for Trust Prediction with Distrust Information.

denoting as $\mathcal{A}_T^p$. Then the performance is assessed as,

$$M_2 = \frac{|\mathcal{A}_T^n \cap \mathcal{A}_T^p|}{|\mathcal{A}_T^n|} \tag{3.5}$$

Experimental Results

We use disTP-m and disTP-s to denote performing multiple steps and a single step distrust propagation in trust propagation (TP), and their comparison results are shown in Figure 3.6. The comparison results of disMF over matrix-factorization based trust prediction (MF) are demonstrated in Figure 3.7. Note that "Random" in figures denotes the performance of randomly guessing.

Let us first examine the performance comparisons when $x = 50$, which are highlighted in Figure 3.6 and Figure 3.7. We make the following observations:

- For the first column results in Figure 3.6, both disTP-s and disTP-m outperform TP. For example, disTP-s obtains 4.28% relative improvement compared to TP. disTP-s and disTP-m incorporate distrust propagation into trust propagation, and the improvement is from distrust propagation. These results support that distrust can improve trust propagation and leads to the performance gain in trust prediction. We also note that most of the time, disTP-s with one-step distrust propagation outperforms disTP-m with multiple step distrust propagation.

	50%	55%	60%	65%	70%	80%	90%
TP	**0.1376**	0.1354	0.1293	0.1264	0.1201	0.1156	0.1098
disTP-s	**0.1435**	0.1418	0.1372	0.1359	0.1296	0.1207	0.1176
disTP-m	**0.1422**	0.1398	0.1359	0.1355	0.1279	0.1207	0.1173
Random	**0.0023**	0.0023	0.0020	0.0019	0.0018	0.0015	0.0013

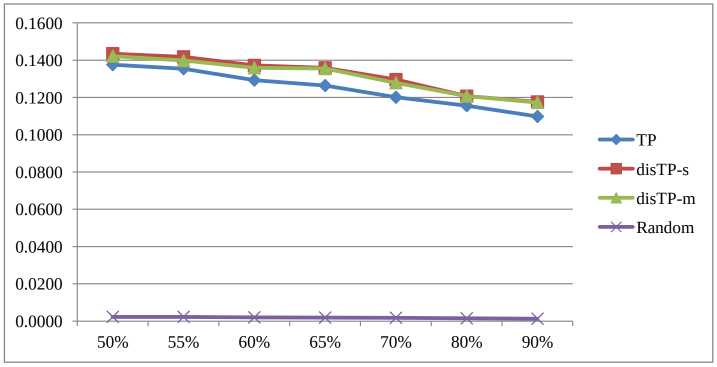


Figure 3.6: Performance Comparison for Trust Propagation without and with Distrust Information.

	50%	55%	60%	65%	70%	80%	90%
MF	**0.1531**	0.1502	0.1489	0.1444	0.1391	0.1332	0.1277
disMF	**0.1665**	0.1654	0.1639	0.1601	0.1563	0.1498	0.1415
Random	**0.0023**	0.0023	0.0020	0.0019	0.0018	0.0015	0.0013

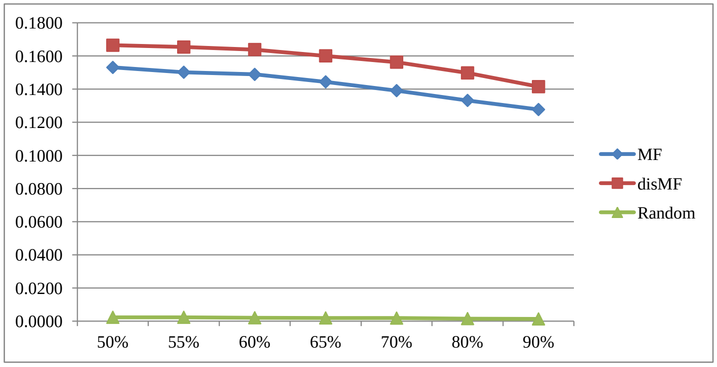


Figure 3.7: Performance Comparison for the Matrix Factorization based Method without and with Distrust Information.

- For the first column results in Figure 3.7, disMF obtains better performance than MF, and gains 8.55% relative improvement over MF.

For other values of x, we have similar observations - distrust can improve the performance of trust prediction. With the help of distrust, we can significantly improve the performance of trust prediction, which suggests that distrust has added value over trust.

3.5 Conclusion

With aforementioned investigations, we can draw a computational understanding of distrust in social media by answering the three questions we asked at the beginning of this chapter. First, the property investigation suggests that distrust presents distinct properties from trust and we cannot extend properties of trust to distrust. Second, the task of distrust prediction with only trust fails to predict distrust by using low trust, which indicates that low trust is not equivalent to distrust and distrust is not the negation of trust. Third, the trust prediction performance is significantly improved with distrust information, which supports that distrust has added value over trust.

Chapter 4

PREDICTING DISTRUST IN SOCIAL MEDIA

It is suggested in research [37, 24] that trust is a desired property while distrust is an unwanted one for an online social community. Intuitively, various online services such as Ciao[1], eBay[2] and Epinions[3] implement trust mechanisms to help users to better use their services, but few of them allow online users to specify distrust relations. To make use of distrust, we need to make them visible on social media sites where distrust does not explicitly present. We propose to predict distrust by mining social media data. Before delving into the discussion of distrust prediction, we delineate its differences from trust/distrust prediction [23] and sign prediction [101] as shown in Figure 4.1. The *distrust prediction* problem in this proposal is quite different from the trust/distrust prediction and sign prediction problems as follows:

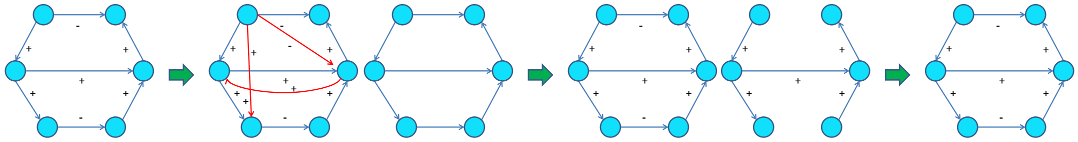
- As shown in Figure 4.1a, the trust/distrust prediction predicts new trust and distrust relations from existing trust and distrust relations. Our prediction problem, as illustrated in Figure 4.1c, assumes that distrust is not accessible in data.

- The sign prediction problem as shown in Figure 4.1b predicts signs of *already existing* relations. The distrust prediction problem needs to identify the pairs of nodes between which distrust relations *are predicted to* exist.

[1] http://www.ciao.co.uk/

[2] http://www.ebay.com/

[3] http://www.epinions.com/



(a) Trust/Distrust Prediction (b) Sign Prediction (c) Distrust Prediction

Figure 4.1: An Illustration of the Differences of Trust/Distrust Link Prediction, Sign Prediction and Distrust Prediction.

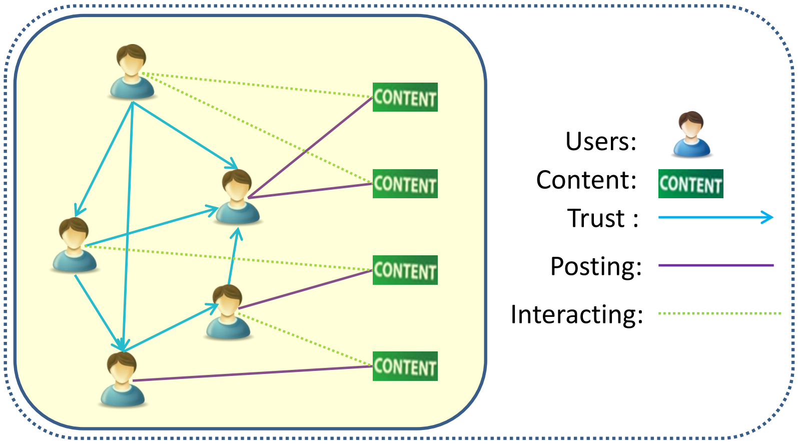


Figure 4.2: An Illustration of Interaction Data in Social Media.

4.1 Problem Statement

To preserve the generality of our approach, it is important to use data which is pervasively available across these social trust systems. Figure 4.2 demonstrates a typical data in social trust systems. First, an obvious source of useful data is trust information which is commonly available in most social trust systems. Second, in most these sites, users can create or post user-generated content and other users can comment, like/dislike and rate such content. For example, in Epinions, users can rate the helpfulness of reviews written by others. In this work, we study the novel problem of distrust prediction from these two pervasive sources in social trust systems - trust information and content-centric interactions.

Let $\mathcal{P} = \{p_1, p_2, \ldots, p_M\}$ be the set of M pieces of user-generated content such as posts. We use $\mathbf{A} \in \mathbb{R}^{n \times M}$ to denote the user-content authorships, where $\mathbf{P}_{ij} = 1$ if p_j is created by u_i, and $\mathbf{P}_{ij} = 0$ otherwise. Users can express opinions on content via comments, likes/dislikes, and ratings. Some sites provide explicit ways of enabling user feedback on content. Examples include likes/dislikes in eBay, and "very helpful"/"not helpful" ratings in Epinions. Other more common forms of feedback allow users to express their opinions in the form of textual comments and replies. In such cases, we adapt off-the-shelf opinion mining tools to extract user opinions from such texts. We use $\mathbf{O} \in \mathbb{R}^{n \times M}$ to represent the user-post opinion relations where $\mathbf{O}_{ij} = 1$, $\mathbf{O}_{ij} = -1$ and $\mathbf{O}_{ij} = 0$, if u_i expresses positive, negative and neutral (or no) opinions, respectively, on p_j.

With the aforementioned notations and definitions, the problem of distrust prediction with trust relations and content-centric user interactions is formally defined as follows:

Given trust relations $\mathbf{T}$*, and content-centric user interactions* $\mathbf{P}$ *and* $\mathbf{O}$*, we aim to develop a predictor* f *to predict distrust relations* $\mathbf{D}$ *with* $\mathbf{T}$*,* $\mathbf{P}$ *and* $\mathbf{O}$ *as,*

$$f: \ \{\mathbf{T}, \mathbf{P}, \mathbf{O}\} \rightarrow \mathbf{D} \tag{4.1}$$

4.2 Data Analysis

Because trust prediction is dependent on "typical" behavior of trust networks such as transitivity and similarity, it is natural to explore similar properties of distrust with respect to trust relations and and content-centric interactions. Such an understanding lays the groundwork for a meaningful distrust prediction model. Note that distrust relations in data analysis only serve as a ground-truth about typical properties and the underlying social theories. However, they will not be explicitly used in the proposed frameworks for the problem of distrust prediction.

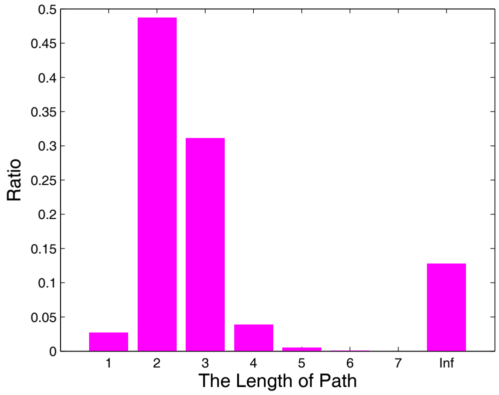


Figure 4.3: Ratio Distribution of the Length of Shortest Path for Pairs with Distrust Relations in the Trust Network.

4.2.1 Where Are our "Foes"?

Our first analytical task is to examine the typical structural relationships of "foes" (or users with distrust relations) within the trust network. In other words, if u_i has a distrust relation to u_j in the distrust network $\mathbf{D}$, we investigate the typical position of u_j with respect to u_i in the trust network $\mathbf{T}$.

For each distrust link u_i-u_j in $\mathbf{D}$, we use breadth-first search to compute the shortest path from u_i to u_j in $\mathbf{T}$. If paths exist from u_i to u_j, we report the length of the shortest path. Otherwise we report the length as "inf" to indicate there is no path from u_i to u_j in $\mathbf{T}$. The ratio distributions of the lengths of the shortest paths for all distrust relations are demonstrated in Figure 4.3. More than 45% of distrust relations u_i-u_j have shortest path lengths less than 3, and more than 80% of them have shortest path lengths less than 4. These results suggest that our "foes" are often close to us in the trust network $\mathbf{T}$. For example, about 82.64% of enemy-pairs are within 3-hops of each other in the trust network of Epinions.

4.2.2 Social Theories in Trust/Distrust Networks

We can view trust/distrust networks as signed networks. In this subsection, we investigate two of the most important social theories for signed networks in trust/distrust networks - balance theory [25] and status theory [23, 39].

For a triad, four possible sign combinations exist - **A**(+,+,+), **B**(+,+,-) **C**(+,-,-) and **D**(-,-,-). Among these four combinations, **A** and **C** are balanced. The way to measure balance of trust/distrust networks is to examine all these triads and then to compute the ratio of **A** and **C** over **A**, **B**, **C** and **D**. We computed the relative presence of these four possible combinations and find that 92.31% triads in Epinions are balanced.

While balance theory is developed for undirected signed networks, for directed signed networks, status theory is introduced in [23, 39]. In status theory, a trust relation from u_i to u_j indicates that u_i has a higher status than u_j; while a distrust relation from u_i to u_j indicates that u_i has a lower status than u_j. For a triad, status theory suggests that if we take each distrust relation, reverse its direction, and flip its sign to trust, then the resulting triangle (with all trust relations) should be acyclic. We first obtain all triads and then follow the aforementioned way to examine whether these triads satisfy status theory or not. We find that 94.73% of triads in Epnions satisfy status theory.

4.2.3 Distrust Relations and Content-centric Interactions

Content-centric interactions relate the opinion of user u_i on the content posted by user u_j. The user u_i can express negative opinions on content posted by another user u_j by disliking, giving negative comments, or negative ratings. Such types of content-centric interactions may be viewed as negative interactions between u_i and

u_j. A negative interaction from u_i to u_j is often a manifestation of user u_i's disagreement and antagonism toward u_j. It is therefore reasonable to surmise that negative interactions might be correlated with distrust relations. In this subsection, we study the correlation between negative interactions and distrust relations.

Let $\mathbf{N} \in \mathbb{R}^{n \times n}$ be a user-user negative interaction matrix where $\mathbf{N}_{ij}$ denotes the number of negative interactions from u_i to u_j. We can obtain $\mathbf{N}$ from the user-content authorship matrix $\mathbf{A}$ and the user-content opinion matrix $\mathbf{O}$ as $\mathbf{N} = -\mathbf{A}(\mathbf{O}^{-})^{\top}$ where $\mathbf{O}^{-} = \frac{\mathbf{O}-|\mathbf{O}|}{2}$ is the negative part of $\mathbf{O}$. To verify the correlation between negative interactions and distrust relations, we aim to answer the following question: Are pairs of users with negative interactions more likely to have distrust relations than those without negative interactions?

For each pair $\langle u_i, u_j \rangle$ with negative interactions (or $\mathbf{N}_{ij} \neq 0$), we first randomly select a user u_k that u_i does not have negative interactions with (or $\mathbf{N}_{ij} = 0$), and then use S (or R) to indicate whether $\langle u_i, u_j \rangle$ (or $\langle u_i, u_k \rangle$) has a distrust relation where $S = 1$ (or $R = 1$) if u_i has a distrust relation to u_j (or u_i has a distrust relation to u_k), otherwise $S = 0$ (or $R = 0$). Let $\mathbf{s}$ be a vector of Ss over all pairs of users with negative interactions and $\mathbf{r}$ be the corresponding vector of Rs. We conduct a two-sample t-test on $\mathbf{s}$ and $\mathbf{r}$. The null hypothesis and the alternative hypothesis are defined as:

$$H_0 : \mathbf{s} \leq \mathbf{r}, \quad H_1 : \mathbf{s} > \mathbf{r}. \tag{4.2}$$

The null hypothesis is rejected at significance level $\alpha = 0.01$ with p-value of 5.72e-89 in Epinions. Evidence from the t-test suggests a positive answer to the question: *there is a strong correlation between negative interactions and distrust relations, and users with negative interactions are likely to have distrust relations.*

We further investigate the direct impact of negative interactions on distrust re-

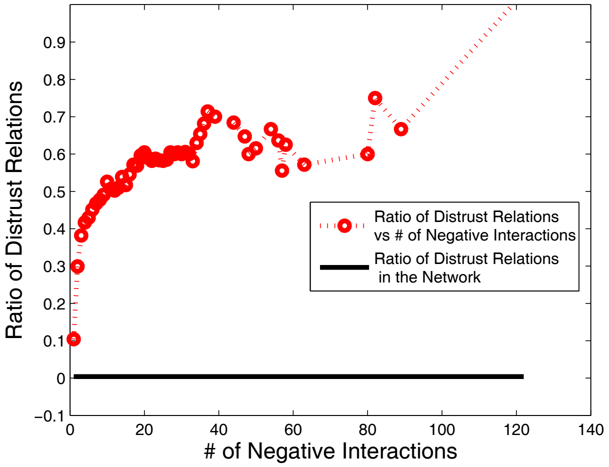


Figure 4.4: Number of Negative Interactions and Distrust Relations.

lations. For a given value of K, we calculated the ratio of pairs with both distrust relations and at least K negative interactions over all pairs with at least K negative interactions. The ratio distributions with respect to the number of negative interactions are demonstrated in Figures 4.4. Note that the ratios of randomly selected pairs with distrust relations among all $n(n-1)$ pairs of users are 2.4177e-04 in Epinions. From the figure, we note that the ratios are much higher than the random ones even when K is very small. This observation further supports the existence of the correlation between negative interactions and distrust relations. Furthermore with increase of K, the ratios tend to increase. Therefore, an increase in the number of negative interactions increases the likelihood of distrust relations between users.

4.2.4 Discussion

We summarize the insights obtained in the aforementioned discussion as follows:

- Most of our “foes” are close to us within a few (e.g., 2 or 3) hops in the trust network;

- Most of triads in trust/distrust networks satisfy balance theory and status theory;
- Pairs of users with negative interactions are more likely to have distrust relations than those without them; and
- Negative interactions between users increase the propensity of distrust relations.

These observations provide the groundwork for the following proposed frameworks for distrust prediction. Algorithms for all variations of trust prediction are either unsupervised methods [23, 77] or supervised methods [43, 59]. In the following two subsections, we will investigate distrust prediction in both unsupervised [80] and supervised scenarios [73].

4.3 Unsupervised Distrust Prediction

Traditional unsupervised trust prediction are usually based on certain properties of trust [87] such as transitivity [23] and low-rank representation [77]. Similarly, we try to model findings and observations of distrust in the last subsection for unsupervised distrust prediction.

4.3.1 Pseudo Distrust Relations

We first divide all $n(n-1)$ pairs of users into three groups $\mathcal{G} = \{\mathcal{G}_1, \mathcal{G}_2, \mathcal{G}_3\}$ as:

- $\mathcal{G}_1$ contains pairs of users with trust relations as:

$$\mathcal{G}_1 = \{\langle u_i, u_j \rangle | \mathbf{T}_{ij} = 1\}; \tag{4.3}$$

- $\mathcal{G}_2$ is the set of pairs of users without trust relations but with negative interactions as:

$$\mathcal{G}_2 = \{\langle u_i, u_j \rangle | \mathbf{T}_{ij} = 0 \wedge \mathbf{N}_{ij} > 0\}; \tag{4.4}$$

- $\mathcal{G}_3$ includes the remaining pairs as:

$$\mathcal{G}_3 = \{\langle u_i, u_j \rangle | \langle u_i, u_j \rangle \notin (\mathcal{G}_1 \cup \mathcal{G}_2)\}. \tag{4.5}$$

$\mathcal{G}_1$ and $\mathcal{G}_2$ correspond to the set of pairs of users with trust relations and negative interactions, respectively. Based on these three groups, we introduce a matrix $\mathbf{F} \in \mathbb{R}^{n \times n}$ to represent user-user trust relations and pseudo distrust relations, and the entities of $\mathbf{F}$ are defined as follows:

- For $\langle u_i, u_j \rangle \in \mathcal{G}_1$, we set $\mathbf{F}_{ij} = 1$ since u_i trusts u_j;
- For $\langle u_i, u_j \rangle \in \mathcal{G}_2$, u_i has negative interactions to u_j, and according to the correlation between negative interactions and distrust relations, u_i is likely to distrust u_j; hence, we assign a pseudo distrust relation from u_i to u_j by setting $\mathbf{F}_{ij} = -1$;
- We do not have evidence of possible relations for $\langle u_i, u_j \rangle \in \mathcal{G}_3$, therefore we set $\mathbf{F}_{ij} = 0$.

the entities of $\mathbf{F}$ are formally defined as follows:

$$\mathbf{F}_{ij} = \begin{cases} 1 & \text{if } \langle u_i, u_j \rangle \in \mathcal{G}_1 \\ -1 & \text{if } \langle u_i, u_j \rangle \in \mathcal{G}_2 \\ 0 & \text{if } \langle u_i, u_j \rangle \in \mathcal{G}_3 \end{cases} . \tag{4.6}$$

The values in $\mathbf{F}$ may be not equally reliable. For example, $\mathbf{F}_{ij}$ for $\langle u_i, u_j \rangle \in \mathcal{G}_1$ is very reliable since we observe trust relations, while values of pairs in $\mathcal{G}_2$ with more negative interactions are more reliable based on the finding - the more negative interactions two users have, the more likely a distrust relation between them exists. Therefore, we define a weight matrix $\mathbf{W} \in \mathbb{R}^{n \times n}$ where $\mathbf{W}_{ij} \in [0, 1]$ is a weight to indicate the reliability of $\mathbf{F}_{ij}$. Next we define the weight matrix as

- We observe trust relations for pairs in $\mathcal{G}_1$; hence for $\langle u_i, u_j \rangle \in \mathcal{G}_1$, we set $\mathbf{W}_{ij} = 1$;

- Our previous finding reveals that the more negative interactions two users have, the more likely a distrust relation between them exists; hence for $\langle u_i, u_j \rangle \in \mathcal{G}_2$, $\mathbf{W}_{ij}$ is defined as a function of the number of negative interactions as $\mathbf{W}_{ij} = g(\mathbf{N}_{ij})$. The function $g(x)$ has following properties - (1) x is a positive integer; (2) $g(x) \in [0, 1]$; and (3) $g(x)$ is non-decreasing function of x; and
- We set $\mathbf{W}_{ij}$ to be a constant $c \in [0, 1]$ for $\langle u_i, u_j \rangle \in \mathcal{G}_3$.

the weight matrix $\mathbf{W}$ is formally defined as,

$$\mathbf{W}_{ij} = \begin{cases} 1 & \text{if } \langle u_i, u_j \rangle \in \mathcal{G}_1 \\ g(\mathbf{N}_{ij}) & \text{if } \langle u_i, u_j \rangle \in \mathcal{G}_2 \\ c & \text{if } \langle u_i, u_j \rangle \in \mathcal{G}_3 \end{cases} . \tag{4.7}$$

4.3.2 An Unsupervised Framework - dTrust

With trust and pseudo distrust relations $\mathbf{F}$ and its weight matrix $\mathbf{W}$, the problem of distrust prediction boils down to a special trust and distrust prediction problem. Therefore we can choose a representative trust and distrust prediction algorithm as our basic algorithm. In this work, we choose the matrix factorization based method introduced in Section 3.4.1. However, we may not apply it to our problem directly since the values in $\mathbf{F}$ may not be reliable. The proposed framework dTrust is based on the new formulation with the user-user trust and pseudo distrust relations $\mathbf{F}$ and its weight matrix $\mathbf{W}$ as,

$$\min_{\mathbf{U},\mathbf{H},\mathbf{r}} \quad \sum_{i=1}^{n}\sum_{j=1}^{n} \big(\mathbf{W}_{ij}(\mathbf{F}_{ij} - \mathbf{U}_i\mathbf{H}\mathbf{U}_j^\top - \lambda r_i r_j)\big)^2 + \alpha(\|\mathbf{U}\|_F^2 + \|\mathbf{H}\|_F^2 + \|\mathbf{r}\|_2^2), \tag{4.8}$$

where the contribution of $\mathbf{F}_{ij}$ to the learning process is controlled by $\mathbf{W}_{ij}$. A large value of $\mathbf{W}_{ij}$, indicating the high reliability of $\mathbf{F}_{ij}$, will force $\mathbf{U}_i\mathbf{H}\mathbf{U}_j^\top$ to tightly fit $\mathbf{F}_{ij}$, while $\mathbf{U}_i\mathbf{H}\mathbf{U}_j^\top$ will loosely approximate $\mathbf{F}_{ij}$ when $\mathbf{W}_{ij}$ is small.

Eq. (4.8) can be rewritten to its matrix form as

$$\min_{\mathbf{U},\mathbf{H},\mathbf{r}} \quad \|\mathbf{W} \odot (\mathbf{F} - \mathbf{U}\mathbf{H}\mathbf{U}^\top - \lambda \mathbf{r}\mathbf{r}^\top)\|_F^2 + \alpha(\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2 + \|\mathbf{r}\|_2^2), \tag{4.9}$$

where $\odot$ is the Hadamard product where $(\mathbf{X} \odot \mathbf{Y})_{ij} = \mathbf{X}_{ij} \times \mathbf{Y}_{ij}$ for any two matrices $\mathbf{X}$ and $\mathbf{Y}$ with the same size. The significance of the introduction of pseudo distrust relations in $\mathbf{F}$ is three fold. First, it enables us to predict distrust relations by incorporating interaction data. Second, it paves a way to model the correlation between negative interactions and distrust relations. Finally the introduction of pseudo distrust relations enables us to exploit social theories from signed networks with trust and pseudo distrust relations, while exploiting social theories in turn may help us mitigate the effects of unreliable values in $\mathbf{F}$, and potentially improves the distrust prediction performance. The optimization problem in dTrust is solved by a decent gradient method and more details are shown in Appendix A.

4.3.3 Evaluation

In this section, we conduct experiments to evaluate the effectiveness of the proposed framework. In particular, we try to answer two questions via experiments - (1) can the proposed framework predict distrust information indirectly with interaction data? and (2) how do the components of dTrust affect its performance? We begin by introducing experimental settings, and then design experiments to seek answers for these questions.

Experimental Settings

Let $\mathcal{A}$ be the set of pairs with trust relations in the Epinions dataset and we sort $\mathcal{A}$ in a chronological order in terms of the time when pairs established trust relations. Assume that there are $x\%$ of pairs in $\mathcal{A}$ establishing trust relations until time t_x. For each x, we collect trust relations, distrust relations, user-review authorship relations and user-review helpfulness ratings until time t_x to form a evaluation dataset *Epinionsx*. In this paper, we vary x as $\{50, 70, 100\}$ and correspondingly we construct three evaluation datasets from the Epinions dataset, i.e., *Epinions50*, *Epinions70* and *Epinions100*. The purpose of varying the values of x is to investigate the performance of the proposed framework on Epinions datasets with different statistics. Other experimental settings and evaluation metric are the same as these in 3.3.2.

Experimental Results

The comparison results are shown in Figure 4.5 and baseline methods in the figure are defined as,

- *negInter*: This method is based on the strong correlation between negative interactions and distrust relations. *negInter* ranks pairs of users based on the numbers of negative interactions. The larger the number of negative interactions, the higher the prediction confidence.

- *random*: this predictor ranks pairs of users randomly. [41] suggests that a random predictor should be used as a baseline method to meaningfully demonstrate the predictor quality since the M1 value is usually low.

From Figure 4.5, we note that the performance of *negInter* is much better than that of *random*, which further demonstrates the existence of a strong correlation between negative interactions and distrust relations. We also observe that the proposed framework dTrust always outperforms baseline methods. Performance comparison between the random predictor and the proposed framework dTrust suggests that dTrust can accurately predict distrust relations by incorporating trust relations and interaction data.

	Epinions50	Epinions70	Epinions100
negInter	0.0546	0.1147	0.1278
dTrust	0.0874	0.1505	0.1739
random	0.00002	0.00004	0.00005

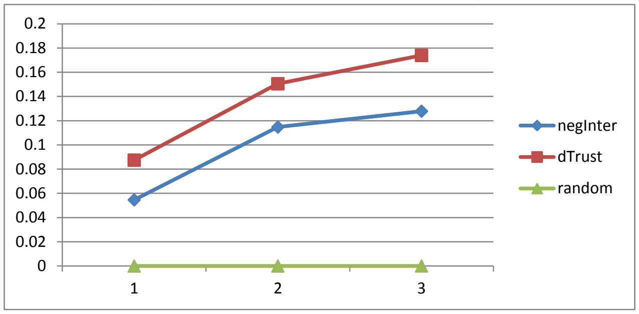


Figure 4.5: Performance Comparison of Different Predictors.

$g(x)$ controls the impact of the number of negative interactions on dTrust and we empirically set $g(x) = 1 - \frac{1}{log(x+1)}$. Next we investigate the importance of the number of negative interactions by studying the performance changes of dTrust with difference choices of $g(x)$ as shown in Table 4.1. Note that "random" in the table denotes that we randomly assign values in $[0, 1]$ to the function. We make the following observations

- When $g(x) = 0$, we eliminate negative interactions and the performance reduces dramatically. This result demonstrates the importance of incorporating interaction data.

- Compared to the performance of $g(x) = 1 - \frac{1}{log(x+1)}$, the performance $g(x)$ with a non-zero constant degrades a lot. These results suggest that modeling the impact of the number of negative interactions on the correlation can improve the performance of dTrust.

- Compared to the performance of $g(x) = 1 - \frac{1}{log(x+1)}$, the performance $g(x)$ with random values also reduces a lot. These results directly suggest that $g(x)$ should not be random values, and further demonstrate the importance of modeling the impact of the number of negative interactions by $\mathbf{W}$.

	Epinions50	Epinions70	Epinions100
$g(x) = 0$	0.00001	0.00003	0.00004
$g(x) = 1$	0.05812	0.11686	0.13039
$g(x) = random$	0.05905	0.11763	0.13207
$g(x) = 1 - \frac{1}{log(x+1)}$	0.08737	0.15054	0.17391

Table 4.1: Difference Definitions of $g(x)$.

We can conclude that (1) $g(x)$ should not be random values; (2) defining $g(x)$ based on the number of negative interactions can significantly improve the performance of dTrust.

4.4 Supervised Distrust Prediction

Traditional supervised methods consider trust prediction as as classification problem [87]. Supervised methods could have several advantages over unsupervised methods such as superior performance, adaptability to different data domains, and variance reduction [42]. In this subsection, we investigate how to tackle the problem of distrust prediction with supervised learning. Similar to traditional trust prediction, we can consider distrust prediction problem as a classification problem where we need to construct labels and extract features. Different from traditional trust prediction, there are unique challenges in preparing training data in the distrust prediction problem. For example, existing relations are given in traditional trust prediction such as trust relations in trust prediction, and trust/distrust relation in trust/distrust prediction, while existing distrust relations are not given in the distrust prediction problem.

4.4.1 Label Construction

Let E_n and E_o denote pairs of users with distrust relations and without any relations, respectively. In most previous formulations of link prediction, including the signed version, label construction is trivial because the presence of links is specified. However, we study the scenario where no distrust relations are provided, and therefore the labels for E_n are really an unspecified subset of $E_o \cup E_n$. What is worse, the sizes of E_n and E_o are extremely imbalanced. For example, the imbalance ratios $E_n : E_o$ are 1:4131 in Epinions. We treat pairs in E_o as positive samples and distrust relations as negative samples. Label construction is to construct positive and negative samples from $E_o \cup E_n$. Since the ratio of E_o in $E_n \cup E_o$ are often bigger than 99.9%, we can randomly select a subset of samples from $E_n \cup E_o$ as positive samples PS. Next we introduce a way to select samples from $E_n \cup E_o$ as negative samples based on previous observations. The process is shown in Algorithm 2.

Next, we describe Algorithm 2 for negative sample construction. The strong correlation between negative interactions and distrust relations suggests that users with negative interactions are likely to have distrust relations. Therefore from line 2 to line 4 in Algorithm 2, we construct negative sample candidates based on this observation. With the trust relations and distrust relations u_i-u_j from NS, we construct a signed network $\mathcal{G}$ in line 5. Most of the triads in signed networks satisfy status theory. Therefore we refine NS by (a) excluding $\langle u_i, u_j \rangle$ from NS if u_i-u_j is in any triads of $\mathcal{G}$ that does not satisfy status theory in line 6; and (b) adding samples $\langle u_i, u_k \rangle$ into NS if u_i-u_k can make all triads that involve u_i and u_k in $\mathcal{G}$ satisfying status theory in line 7. The reliability of these negative samples may vary. For example, observations from data analysis indicate that negative sample candidates with more negative interactions are more likely to have distrust links, and are therefore more likely to be

Algorithm 2 Negative Sample Construction

```
Input : The trust network $\mathbf{T}$ and user-user negative interaction matrix $\mathbf{N}$
Output : Negative sample set $NS$ and the reliability weight matrix $\mathbf{W}$
1: Initialize $NS = \emptyset$
2: for all $\mathbf{N}_{ij} \neq 0$ do
3:     $NS = NS \cup \{\langle u_i, u_j \rangle\}$
4: end for
5: Construct $\mathcal{G}$ as a signed network with trust relations from $\mathbf{T}$ and distrust relations
   $u_i$-$u_j$ from $NS$
6: Remove samples $\langle u_i, u_j \rangle$ from $NS$ if $u_i$-$u_j$ is in any triads of $\mathcal{G}$ that does not
   satisfy status theory
7: Add samples $\langle u_i, u_k \rangle$ into $NS$ if $u_i$-$u_k$ can make all triads that involve $u_i$ and $u_k$
   in $\mathcal{G}$ satisfying status theory
8: for all $\langle u_i, u_j \rangle \in NS$ do
9:     Calculate a reliability weight $\mathbf{W}_{ij}$
10: end for
```

reliable. Therefore, we associate each $\langle u_i, u_j \rangle$ with a reliability weight $\mathbf{W}_{ij}$, which is defined as follows:

$$\mathbf{W}_{ij} = \begin{cases} g(\mathbf{N}_{ij}) & \text{if } \mathbf{N}_{ij} \neq 0 \\ r & \text{otherwise} \end{cases} . \tag{4.10}$$

if the pair $\langle u_i, u_j \rangle \in NS$ has negative interactions, we define the reliability weight as a function g of the number of negative interactions $\mathbf{N}_{ij}$. Otherwise, the pair $\langle u_i, u_j \rangle \in NS$ is added by line 7 in Algorithm 2 and we set the reliability weight to a constant r.

4.4.2 Feature Extraction

We extract three types of features corresponding to user features, pair features and sign features. User features and pair features are extracted from two given sources, such as trust relations and content-centric interactions, as follows:

- User features are extracted for each user u_i including u_i's indegree (or outdegree) in terms of trust relations, the number of triads that u_i involved in, the number of content-centric items (e.g., posts) that u_i creates, the number of u_i's posts that obtain positive (or negative) opinions, and the number of positive (or negative) opinions u_i expresses; and

- Pair features are extracted for a pair of users $\langle u_i, u_j \rangle$ including the number of positive (or negative) interactions from u_i to u_j, the number of positive (or negative) interactions from u_j to u_i, Jaccard coefficients of indegree (or outdegree) of u_i and u_j in terms of trust relations, and the length of the shortest path between u_i and u_j.

We construct a weighted signed network with the given trust relations and distrust relations from NS where the weights of trust relations are 1 and the weights of distrust relations are their reliability weights. For a pair $\langle u_i, u_j \rangle$, signed features include weighted indegree (or outdegee) in terms of trust relations of u_i, weighted indegree (and outdegee) in terms of distrust relations of u_j, Jaccard coefficients of indegree (or outdegree) of u_i and u_j in terms of distrust relations and 16 weighted triads suggested by [38].

With definitions of user features, pair features and sign features, we extract 45 features in total for each pair $\langle u_i, u_j \rangle$ including 8 user features of u_i, 8 user features of u_j, 7 pair features, and 22 signed features.

4.4.3 A Supervised Framework - NeLP

Through label construction and feature extraction, we prepare training data to learn classifiers for the distrust prediction problem. However, the labels of the training data are noisy, and especially so for negative samples. Therefore, it is necessary for the base classifier to be tolerant to training data noise. In this paper, we choose a soft-margin version of support vector machines as our basic classifier because it has been proven to be highly noise-tolerant [14].

Let $\mathcal{X} = \{x_1, x_2, \ldots, x_N\}$ be the set of user pairs in $E_o \cup E_n$ and $\mathbf{x}_i$ be the feature vector representation of the pair x_i. The standard soft-margin support vector machine for the distrust prediction problem is as follows:

$$\begin{aligned} \min_{\mathbf{w},b,\epsilon} \quad & \frac{1}{2}\|\mathbf{w}\|_2^2 + C \sum_{x_i \in PS \cup NS} \epsilon_i \\ s.t. \quad & y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 - \epsilon_i, \;\; x_i \in PS \cup NS \\ & \epsilon_i \geq 0 \;\; x_i \in PS \cup NS \end{aligned} \tag{4.11}$$

Eq. (4.11) introduces the term ϵ_i for the soft-margin slack variable of x_i, which can be viewed as the allowance for the noise in this training sample. The parameter C controls the degree of impact of this term. In the distrust prediction problem, the noise-levels of positive and negative samples are different because positive samples PS are generally more robust than the (indirectly derived) negative samples. As discussed earlier, the reliability of negative samples is explicitly quantified with their weights. These intuitions suggest that we should allow more errors in negative samples especially when their weights suggest unreliability. This yields the following formulation:

$$\begin{aligned} \min_{\mathbf{w},b,\epsilon} \quad & \frac{1}{2}\|\mathbf{w}\|_2^2 + C_p \sum_{x_i \in PS} \epsilon_i + C_n \sum_{x_j \in NS} c_j \epsilon_j \\ s.t. \quad & y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 - \epsilon_i, \quad x_i \in PS \\ & y_j(\mathbf{w}^\top \mathbf{x}_j + b) \geq 1 - \epsilon_j, \quad x_j \in NS \\ & \epsilon_i \geq 0, \ \epsilon_j \geq 0 \end{aligned} \tag{4.12}$$

In Eq. (4.12), we use two parameters C_p and C_n to weight the positive and negative errors differently. We use a larger value for C_p compared to C_n to reflect the differential behavior of the positive and negative samples. For a negative sample x_j, we introduce a weight c_j to further control its error based on its quantified reliability weight. For the negative sample x_j corresponding to the pair $\langle u_i, u_k \rangle$, we set $c_j = \mathbf{W}_{ik}$ where $\mathbf{W}_{ik}$ is the reliability weight for $\langle u_i, u_k \rangle$. This additional term allows differential control of the noise in negative samples of varying reliability.

Balance theory suggests that triads in signed networks are likely to be balanced; hence we want to maintain or increase the structural balance after distrust prediction. If there is a trust relation between u_i and u_j, and both u_i and u_j do not have trust relations with another user u_k, the types of (u_i, u_k) and (u_j, u_k) in the distrust graph $\mathcal{G}_n$ are likely to be the same. In other words, to maintain or increase the structural balance, it is likely that both are distrust relations where $\langle u_i, u_j, u_k \rangle$ forms a balanced triad or both are missing relations where there is no triad among $\langle u_i, u_j, u_k \rangle$. With this intuition, we introduce a matrix $\mathbf{B}$ where $\mathbf{B}_{h\ell} = 1$ if there is a trust relation between u_i and u_j where we assume that x_h and x_ℓ denote pairs $\langle u_i, u_k \rangle$ and $\langle u_j, u_k \rangle$, respectively. Otherwise, we assume that $\mathbf{B}_{h\ell} = 0$. Then, we force x_h and x_ℓ to have the same types of links if $\mathbf{B}_{h\ell} = 1$ by introducing a balance-theory regularization:

$$\min \ \frac{1}{2} \sum_{h,\ell} \mathbf{B}_{h\ell} (\mathbf{w}^\top \mathbf{x}_h - \mathbf{w}^\top \mathbf{x}_\ell)_2^2 = \mathbf{w}^\top \mathbf{X} \mathcal{L} \mathbf{X}^\top \mathbf{w} \tag{4.13}$$

Here, $\mathcal{L}$ is the Laplacian matrix based on $\mathbf{B}$. The number of pairs in $E_n \cup E_o$ is usually very large, which leads to a large number of terms in the balance theory regularization. The observation from data analysis suggests that our "foes" are usually close to us in the trust network. Hence, in this work, we only consider pairs whose shortest path lengths are 2, and pairs in NS and PS in the balance theory regularization. We assume that there are $l+\mu$ samples in $\mathbf{X}$ where the first l ones are from $PS \cup NS$. The significance of the introduction of the balance theory regularization is two-fold. First, it allows us to model balance theory. Second, it allows us to include more samples during the learning process in addition to NS and PS. A similar function is achieved by this approach, as achieved by unlabeled samples in semi-supervised learning [106]. With these components, the proposed NeLP framework is able to solve the following optimization problem:

$$\begin{aligned}
\min_{\mathbf{w},b,\epsilon} \quad & \frac{1}{2}\|\mathbf{w}\|_2^2 + C_p \sum_{x_i \in PS} \epsilon_i + C_n \sum_{x_j \in NS} c_j \epsilon_j + \frac{C_b}{2} \mathbf{w}^\top \mathbf{X} \mathcal{L} \mathbf{X}^\top \mathbf{w} \\
s.t. \quad & y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 - \epsilon_i, \quad x_i \in PS \\
& y_j(\mathbf{w}^\top \mathbf{x}_j + b) \geq 1 - \epsilon_j, \quad x_j \in NS \\
& \epsilon_i \geq 0, \;\; \epsilon_j \geq 0 \qquad (4.14)
\end{aligned}$$

We solve the optimization problem in Eq. (4.14) based on the dual form [4] and more details are presented in Appendix B.

4.4.4 Evaluation

In this subsection, we present experiments which (a) quantify the performance of the proposed NeLP framework in predicting distrust links, and (b) evaluate the contribution of various model components to the performance. We begin by introducing performance evaluation metrics, which are useful in both contexts.

	True class = -1	True class = 1
Predicted class = -1	true pos. (tp)	false pos. (fp)
Predicted class = 1	false neg. (fn)	true neg. (tn)

Table 4.2: Confusion Matrix of a Binary Classifier.

Experimental Settings

Distrust prediction can be viewed as a highly imbalanced classification problem. In such case, straightforward accuracy measures are well known to be misleading [96]. For example, the trivial classifier that labels all samples as missing relation can have a 99.99% accuracy in Epinions. In distrust prediction, we aim to achieve high precision and recall over distrust relations, defined in terms of the confusion matrix of a classifier as shown in Table 4.2: $precision = \frac{tp}{tp+fp}$ and $recall = \frac{tp}{tp+fn}$. Usually precision and recall are combined into their harmonic mean, the F-measure. Therefore we will adopt F1-measure as one metric for the performance evaluation. As suggested in [96], in some scenarios, we put more emphasis on precision because the most challenging task is to seek some distrust relations with high probability, even at the price of increasing false negatives. Hence, we also report the precision performance.

Performance of Distrust Prediction

For the evaluation purpose, we define the following baseline methods for the proposed framework NeLP:

- *Random*: This predictor randomly guesses the labels of samples.
- *sPath*: Observations in data analysis suggest that our "foes" are always close to us in the trust network and *sPath* assigns distrust relations to pairs whose shortest path lengths is L;

Algorithms	Epinions	
	F1	Precision
random	0.0005	0.0002
sPath	0.0040	0.0075
negIn	0.2826	0.2097
negInS	0.2893	0.2124
NeLP-negIn	0.3206	0.2812
NeLP	0.3242	0.2861

Table 4.3: Performance Comparison of Distrust Prediction in Epinions.

- *negIn*: Given the strong correlation between negative interactions and distrust links, *negIn* suggests distrust relations to these pairs with negative interactions;
- *negInS*: after obtaining distrust candidates via *negIn*, *negInS* further refines these candidates by performing a removing step and an adding step as shown in Algorithm 2; and
- *NeLP-negIn*: NeLP-negIn is a variant of the proposed NeLP framework. Instead of using distrust links suggested by *negInS* as NeLP, NeLP-negIn uses distrust relations found by *negIn*.

For parameterized methods, we report the best performance of each baseline method. For NeLP, we set its parameters as $\{C_p = 1, C_n = 0.5, C_b = 0.1\}$. We empirically find that $g(x) = 1 - \frac{1}{\log(1+x)}$ works well for the proposed framework. More details about parameter sensitivity of NeLP will be discussed later. The comparison results are demonstrated in Table 5.1.

We make the following observations:

- *sPath* obtains much better performance than random guessing, which further supports the hypothesis that our "foes" are close to us in the trust network;
- *negIn* improves the performance significantly. These results suggest the existence of correlation between negative interactions and distrust relations;
- by removing candidates suggested by *negIn* that do not satisfy status theory and adding candidates to make open triads closure to satisfy status theory, *negInS* outperforms *negIn*. For example, *negInS* gains 2.37% relative improvement in terms of F1-measure in Epinions. These results indicate that status theory can help us remove some noisy samples and add some useful samples for training. These observations can also be used to explain the reason why the performance of NeLP based on distrust relations suggested by *negInS* is better than that based on *negIn*; and
- The proposed framework always obtains the best performance.

Component Analysis of NeLP

There are three important components of NeLP. First, NeLP introduces C_n to control errors in negative samples. Second, NeLP introduces c_j to control the error in the sample x_j, which is related to the number of negative interactions based on our observations from data analysis. Third, NeLP introduces balance theory regularization to model balance theory, which also allows us to include more samples in the classifier learning process. Next we discuss the effects of these components. By setting $C_p = 1$ and varying different values of C_n, c_j and C_b, we can examine the impact of these components on the performance of NeLP. The results of component analysis are shown in Tables 4.4.

C_n	c_j	C_b	F1-measure	Precision
0.5	$f(x) = 1 - \frac{1}{\log(1+x)}$	0.1	0.3242	0.2861
1	$f(x) = 1 - \frac{1}{\log(1+x)}$	0.1	0.3188	0.2793
0.5	f(x) = 1	0.1	0.3067	0.2612
0.5	$f(x) = 1 - \frac{1}{\log(1+x)}$	0	0.3084	0.2686
1	f(x) = 1	0	0.2992	0.2342

Table 4.4: Component analysis for NeLP in Epinions.

The first row in the table represents the performance of NeLP with all three components. We make the following observations about different variations of NeLP in other rows of the table:

- in the second row, we set $C_n = 1$, which gives equal weights to positive and negative samples. This approach effectively eliminates the *differential* importance given to errors from negative samples. The performance degrades, which suggests that the errors of negative and positive samples should be treated differently;

- in the third row, we set $c_j = 1$ instead of the reliability weight related to the number of negative interactions to eliminate the component controlling the error in the negative sample x_j. The performance reduces a lot. For example, the precision reduces by 8.70% in Epinions. These results support the importance of the number of negative interactions to indicate the reliability of negative samples;

- in the fourth row, we set $C_b = 0$ to eliminate the contribution from the balance theory regularization. The performance is consistently worse than that with the balance theory regularization. This illustrates the importance of the balance theory regularization in the proposed NeLP framework; and

- in the fifth row, we eliminate all these three components and the performance further degrades. These results suggest that the three components contain complementary information.

4.5 Conclusion

In this chapter, we propose an unsupervised framework dTrust and a supervised framework NeLP to predict distrust by leveraging trust and content-centric user interactions. We make a number of findings about distrust including (1) our distrusted users are close to us (usually within 2 or 3 hops); (2) most triads in trust/distrust networks satisfy balance and status theories; (3) there is a strong correlation between distrust and negative interactions; and (4) the more negative interactions two users have, the more likely a distrust relation exists between them. Evaluations of dTrust and NeLP on Epinions suggest that (1) balance and status theories play important roles in distrust prediction; (2) negative interactions are strong indicators of distrust; and (3) distrust can be accurately predicted by using trust and content-centric user interactions.

Chapter 5

APPLYING DISTRUST

Trust is used in many real-world applications such as node classification [65], information propagation [31], recommendation [82, 91], information filtering [76] and feature selection [84, 85, 88, 86]. Distrust plays a different role in many ways from trust. Simply extending applications of trust may not work for distrust [10]. In addition, it would be ideal to put research findings into real-world applications. It is also an ultimate assessment of impact for distrust. Hence, we propose to *apply distrust together with trust* in our efforts to further understand the role and added value of distrust. We focus on two applications of distrust - node classification and recommendation. The reason is two-fold. First, these two problems are quite general and many real-world applications can be essentially considered as one of these two problems. For example, sentiment analysis can be formulated as a classification problem [70, 26, 90]. Second, these two applications can serve as examples when exploring other data-intensive applications.

5.1 Node Classification

User information such as demographic values, interest, beliefs or other characteristics plays an important role in helping social media sites provide better services for their users such as recommendations and content filtering. However, most social media users do not share too much of their information [103]. For example, more than 90% of users in Facebook do not reveal their political views [1]. One way of bridging this knowledge gap is to infer missing user information by leveraging the pervasively available network structures in social media. An example of such inference is that of node classification in trust networks. The node classification problem has been extensively studied in the literature [19]. Existing node classification algorithms can be mainly grouped into local classifier based methods and random walk based methods [5]. The vast majority of these algorithms have focused on trust networks [58, 92, 44, 106, 104, 65], while little work exists for trust/distrust networks [71].

5.1.1 Problem Statement

Let $\mathcal{C} = \{c_1, c_2, \ldots, c_m\}$ be the set of m label classes. Assume that $\mathcal{U}^L = \{u_1, u_2, \ldots, u_N\}$ is the set of N labeled nodes where $N < n$ and $\mathcal{U}^U = \mathcal{U} \backslash \mathcal{U}^L$ is the set of $n - N$ unlabeled users. We use $\mathbf{Y} \in \mathbb{R}^{N \times m}$ to denote the label indicator matrix for $\mathcal{U}^L$ where $\mathbf{Y}_{ik} = 1$ if u_i is labeled as c_k, and $\mathbf{Y}_{ik} = 0$ otherwise. With the aforementioned notations and definitions, the problem of node classification in a trust/distrust can be formally stated as follows:

Given a trust/distrust network $\mathcal{G}$ with trust relations $\mathbf{T}$, distrust relations $\mathbf{D}$, and labels $\mathbf{Y}$ for some nodes $\mathcal{U}^L$, the problem of node classification in a trust/distrust network aims to infer labels for the unlabeled nodes $\mathcal{U}^U$.

5.1.2 Transforming Algorithms from Trust to Trust and Distrust Networks

Existing node classification algorithms in trust networks can be mainly divided into local classifier based methods and random walk based methods [5]. In this subsection, we investigate how to generalize these key categories of representative algorithms in trust networks to trust/distrust networks.

Local Classifier based Methods

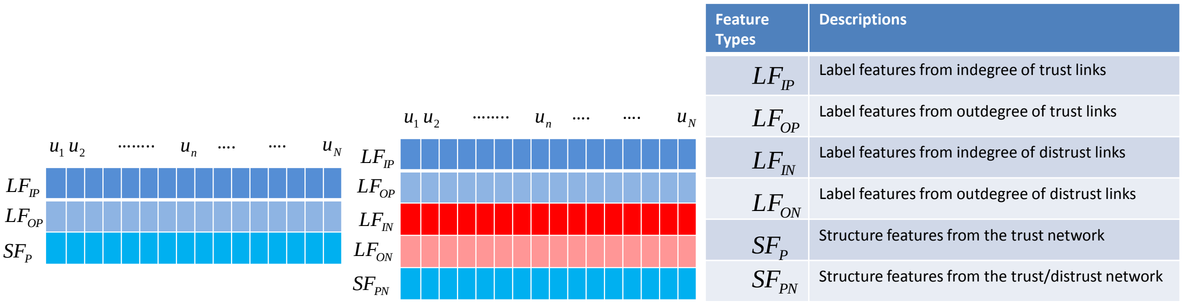
Local classifier based methods use local neighborhood information to learn local classifiers [58, 44, 49]. Iterative classification methods (or ICA) [44] are representative methods in this family. A typical ICA algorithm first extracts feature vectors for nodes. Typically, three types of features are extracted as shown in Figure 5.1a:

- $LF_{IP} = \{f_1, f_2, \ldots, f_m\}$ denotes the set of features of label distributions of incoming relations (or indegree) of the trust network $\mathbf{T}$. The number of features in LF_{IP} is equal to that of class labels and the value of f_k for u_j is often calculated as the frequency of c_k in the incoming relations of u_j in $\mathbf{T}$;
- Similarly, LF_{OP} is the set of features of label distributions of outgoing relations (or outdegree) of $\mathbf{T}$; and
- SF_P is the set of features extracted from the topological information of $\mathbf{T}$ such as indegree, outdegree,and the number of triads.

With the constructed features, it builds a traditional classifier with labeled nodes $\mathcal{U}^L$. The resulting classifier is used to infer labels of nodes in $\mathcal{U}^U$ in an iterative fashion, where the most confidently predicted labels are added to the labeled set. In each iteration a new set of features LF_{IP} and LF_{OP} is extracted with the augmented labels, and the aforementioned process is repeated until stable performance is achieved or a maximum number of iterations have been executed.

To transform the aforementioned ICA algorithm to trust/distrust networks, we define the extracted features from trust/distrust networks as shown in Figure 5.1b:

- We continue to use the feature sets LF_{IP} and LF_{OP} because a trust/distrust network $\mathcal{G}$ naturally includes a social network with only trust relations $\mathbf{T}$;
- Similar to LF_{IP} for trust relations, we define LF_{IN} to capture label distributions of incoming distrust relations (or indegree of trust relations);
- Similar to LF_{OP} for trust relations, LF_{ON} is defined to capture label distributions of outgoing distrust relations (or outdegree of distrust relations); and


u_1 u_2 .... u_n .... u_N
LF_{IP}
LF_{OP}
SF_P
u_1 u_2 .... u_n .... u_N
LF_{IP}
LF_{OP}
LF_{IN}
LF_{ON}
SF_{PN}

Feature Types	Descriptions
LF_{IP}	Label features from indegree of trust links
LF_{OP}	Label features from outdegree of trust links
LF_{IN}	Label features from indegree of distrust links
LF_{ON}	Label features from outdegree of distrust links
SF_P	Structure features from the trust network
SF_{PN}	Structure features from the trust/distrust network

(a) Trust Networks (b) Trust/Distrust Networks (c) Feature Descriptions

Figure 5.1: Feature Construction for Iterative Classification Methods in Trust and Trust/Distrust Networks.

- We redefine structural features SF_{PN} for trust/distrust networks according to one of the most important theories for trust/distrust networks, i.e., balance theory [25]. According to balance theory, ℓ-length circles can capture important topological properties of trust/distrust [11] and we extract ℓ-length circles for SF_{PN}. For example, when $\ell = 3$, SF_{PN} is the set of triad features suggested by [38].

A summary of the extracted features for ICA in trust and trust/distrust networks is illustrated in Figure 5.1. To transform ICA, we define two feature sets LF_{IN} and LF_{ON} based on distrust relations, and redefine structural features SF_{PN} according to balance theory.

Random Walk based Methods

Random walk based methods propagate labels from labeled nodes to unlabeled nodes by performing random walks on the trust network [6, 106, 104, 105]. Graph regularization based methods are representative methods in this family [5]. A typical formulation for graph regularization based methods is as follows [105]:

$$\min \sum_{u_i,u_j \in \mathcal{U}} \mathbf{T}_{ij}\left(\frac{\mathbf{L}_i}{\sqrt{d_i^{IP}}} - \frac{\mathbf{L}_j}{\sqrt{d_j^{OP}}}\right)^2 + \mu \sum_{u_k \in \mathcal{U}^L} \|\mathbf{L}_k - \mathbf{Y}_k\|_2^2 \tag{5.1}$$

where $\mathbf{L}_i$ is the predicted label indicator vector for u_i in the network $\mathcal{G}$, and d_i^{IP} and d_j^{OP} are the indegree and outdegree of trust relations for u_i and u_j, respectively. In Eq.(5.1), the first term ensures greater local consistency of labels of users connected with trust relations, and the second term enforces global consistency of the inferred labels with the provided training labels. The balance between the two criteria is controlled by a parameter μ.

Next, we discuss two possible ways of generalizing the aforementioned algorithm to the trust/distrust scenario:

- The underlying assumption of the aforementioned algorithm is that two users with a trust relation are likely to share similar labels, which can be explained by two popular social theories, i.e., homophily and social influence. Intuitively, a distrust relation may denote dissimilarity or distance. Therefore, one possible way to capture distrust relations is to force labels of two users with a distrust relation dissimilar by introducing a term in Eq. (5.1) as:

$$\begin{aligned} \min & \sum_{u_i,u_j \in \mathcal{U}} \mathbf{T}_{ij}\left(\frac{\mathbf{L}_i}{\sqrt{d_i^{IP}}} - \frac{\mathbf{L}_j}{\sqrt{d_j^{OP}}}\right)^2 + \mu \sum_{u_k \in \mathcal{U}^L} \|\mathbf{L}_k - \mathbf{Y}_k\|_2^2 \\ & - \eta \sum_{u_i,u_j \in \mathcal{U}} \mathbf{D}_{ij}\left(\frac{\mathbf{L}_i}{\sqrt{d_i^{IN}}} - \frac{\mathbf{L}_j}{\sqrt{d_j^{ON}}}\right)^2 \end{aligned} \tag{5.2}$$

  where d_i^{IN} and d_j^{ON} are the indegree and outdegree of distrust relations for u_i and u_j, respectively, and the third term captures the contribution from distrust relations, which is controlled by η.

- Previous work demonstrated that more than 90% of triads in trust/distrust networks satisfy status theory [39]. To apply Eq. (5.1), we can convert distrust relations to trust links via status theory.

Let $S_{ij} = 1$ if u_i has a trust relation to u_j and $S_{ij} = -1$ for a distrust relation. In [38], an approach based on status theory is proposed to determine the sign from u_i to u_k in an open triad with given signs between u_i and u_j, and between u_j and u_k. The algorithm first flips the directions of relations between u_i and u_j, and between u_j and u_k, if possible, so that they point from u_i to u_j and u_j to u_k. Then it also flips the signs of the relations correspondingly if we flip their directions. Finally the sign from u_i to u_k is $S_{ik} = S_{ij} + S_{jk}$.

According to the aforementioned algorithm, we can convert distrust relations into trust relations. Three examples are shown in Figure 5.2 where $u_k \rightarrow u_i$ in A, $u_i \rightarrow u_k$ in B and $u_k \rightarrow u_i$ in C are converted trust relations. We have $S_{ij} = -1$ and $S_{jk} = -1$ for the open triad A in Figure 5.2 and next we will use A as an example to illustrate how to covert distrust relations to trust relations. According to the algorithm, we can calculate that $S_{ki} = 1$; hence, we add a trust relation from u_k to u_i. A similar process can be performed for B and C. After that, we remove all distrust relations and the original trust/distrust network is converted into a trust network. As a result, Eq. (5.1) will be applicable to this new trust network.

5.1.3 The Proposed Framework - NCSSN

In this section, we will first introduce a node classification algorithm with only trust relations, and then give details about how to capture distrust relations and the proposed NCSSN framework.

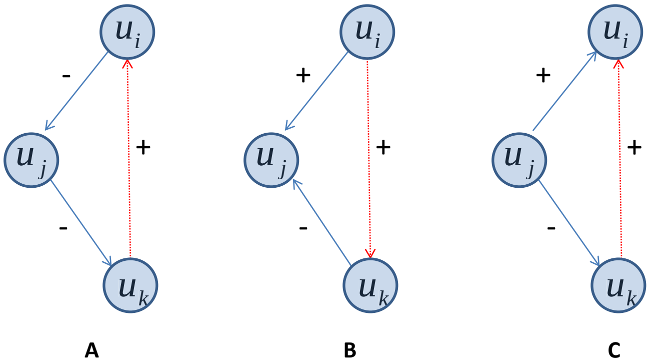


Figure 5.2: Examples of Converting Distrust Relations into Trust Relations according to Status Theory.

A Node Classification Algorithm with Only Trust Relations

To perform the classification, we associate latent factors with nodes. Let $\mathbf{U}_i \in \mathbb{R}_+^{1\times K}$ be the latent factor of u_i. Then, a trust relation from u_i to u_j can be modeled as the interaction between their latent factors as $\mathbf{T}_{ij} = \mathbf{U}_i\mathbf{H}^t\mathbf{U}_j^\top$ where $\mathbf{H}^t \in \mathbb{R}_+^{K\times K}$ captures the interaction for trust relations [77]. We use a linear classifier $\mathbf{W} \in \mathbb{R}^{K\times m}$ to capture the label information from labeled nodes based on their latent factors as $\mathbf{Y}_i = \mathbf{U}_i\mathbf{W}$. The proposed node classification algorithm with only trust relations solves the following optimization problem:

$$\min_{\mathbf{H}^p\geq 0,\mathbf{U}\geq 0,\mathbf{W}} \sum_{u_i,u_j\in\mathcal{U}} \|\mathbf{T}_{ij} - \mathbf{U}_i\mathbf{H}^t\mathbf{U}_j^\top\|_2^2 + \alpha \sum_{u_i\in\mathcal{U}^L} \|\mathbf{U}_i\mathbf{W} - \mathbf{Y}_i\|_2^2 \tag{5.3}$$

where α controls the contribution from labeled nodes.

The Eq. (5.3) can be rewritten in matrix form as follows:

$$\min_{\mathbf{H}^p\geq 0,\mathbf{U}\geq 0,\mathbf{W}} ||\mathbf{T} - \mathbf{U}\mathbf{H}^t\mathbf{U}^\top\|_2^2 + \alpha\|\mathbf{C}(\mathbf{U}\mathbf{W} - \mathbf{Y})\|_F^2 \tag{5.4}$$

where $\mathbf{U} = [\mathbf{U}_1; \mathbf{U}_2; \ldots; \mathbf{U}_n] \in \mathbb{R}_+^{n\times K}$ and $\mathbf{C} \in \mathbb{R}^{n\times n}$ is a diagonal matrix, where $\mathbf{C}_{ii} = 1$ if $u_i \in \mathcal{U}^L$ and $\mathbf{C}_{ii} = 0$, otherwise.

Capturing Distrust Relations

Independent information from distrust relations is crucial to account for distinct topological properties of distrust relations in modeling [69]; while as suggested in [38], trust and distrust relations should also be viewed as tightly related features in a single structure. Therefore, we capture two types of information when modeling distrust relations. One is information from *only* distrust relations which we refer to as independent information from distrust relations. The other is information derived from the *interactions* between trust and distrust relations which we refer to as dependent information from distrust relations.

To capture independent information, which is similar to modeling trust relations, a distrust relation from u_i to u_j can be modeled as the interaction between their latent factors as follows:

$$\mathbf{D}_{ij} = \mathbf{U}_i \mathbf{H}^d \mathbf{U}_j^\top \tag{5.5}$$

Here, the notation $\mathbf{H}^d \in \mathbb{R}^{K \times K}$ is introduced to capture the interaction for distrust relations.

The notion of structural balance was extended by [15]. According to this principle, a structure in a trust/distrust network should ensure that users are able to have their "friends" closer than their "foes" i.e., users should sit closer to their "friends" (or users with trust relations) than their "foes" (or users with distrust relations). For $\langle i, j, k \rangle$ where u_i has a distrust relation to u_j and a distrust relation to u_k, we force u_i closer to her "friend" u_j than her "foe" u_k in terms of their latent factors to capture dependent information according to the aforementioned theory. To achieve this goal, we consider the following two cases:

- *Case 1:* If a user u_i sits closer to her "friend" u_j than her "foes" u_k, i.e., $\|\mathbf{U}_i - \mathbf{U}_j\|_2^2 - \|\mathbf{U}_i - \mathbf{U}_k\|_2^2 < 0$, we should not penalize this case since we expect it.

- *Case 2:* If a user u_i sits closer to her "enemy" u_k than her "friend" u_j, i.e., $\|\mathbf{U}_i - \mathbf{U}_j\|_2^2 - \|\mathbf{U}_i - \mathbf{U}_k\|_2^2 > 0$, we should add a penalty to pull u_i closer to u_j than u_k.

Based on the aforementioned analysis, we propose the following formulation to capture dependent information from distrust relations as

$$\min \sum_{\langle i,j,k \rangle \in \mathcal{S}} max(0, \|\mathbf{U}_i - \mathbf{U}_j\|_2^2 - \|\mathbf{U}_i - \mathbf{U}_k\|_2^2) \tag{5.6}$$

$\mathcal{S}$ is the set of $\langle i, j, k \rangle$ where u_i has a trust relation to u_j while has a distrust relation to u_k, which is formally defined as

$$\mathcal{S} = \{\langle i, j, k \rangle | \mathbf{A}_{ij}^p = 1 \wedge \mathbf{A}_{ik}^n = 1\} \tag{5.7}$$

Next, we give details on the inner workings of Eq. (5.6):

- When a user u_i sits closer to her "friend" u_j than her "foe" u_k, i.e., *Case 1*, the minimizing term in Eq. (5.6) is 0 since $\|\mathbf{U}_i - \mathbf{U}_j\|_2^2 - \|\mathbf{U}_i - \mathbf{U}_k\|_2^2 < 0$. We do not add any penalty for *Case 1*;

- When a user u_i sits closer to her "foe" u_k than her "friend" u_j, i.e., *Case 2*, the minimizing term in Eq. (5.6) is $\|\mathbf{U}_i - \mathbf{U}_j\|_2^2 - \|\mathbf{U}_i - \mathbf{U}_k\|_2^2$ since $\|\mathbf{U}_i - \mathbf{U}_j\|_2^2 - \|\mathbf{U}_i - \mathbf{U}_k\|_2^2 > 0$. Eq. (5.6) will pull u_i closer to u_j than u_k for *Case 2*.

Eq. (5.6) can be rewritten in the matrix form as follows:

$$\sum_{\langle i,j,k \rangle \in \mathcal{S}} max(0, \|\mathbf{U}_i - \mathbf{U}_j\|_2^2 - \|\mathbf{U}_i - \mathbf{U}_k\|_2^2) = \sum_{\langle i,j,k \rangle \in \mathcal{S}} f_{ijk} Tr(\mathbf{M}^{ijk} \mathbf{U}\mathbf{U}^\top) \tag{5.8}$$

where $\mathbf{M}^{ijk}$ has $\mathbf{M}^{ijk}_{ij} = \mathbf{M}^{ijk}_{ji} = \mathbf{M}^{ijk}_{kk} = -1$ and $\mathbf{M}^{ijk}_{ik} = \mathbf{M}^{ijk}_{ki} = \mathbf{M}^{ijk}_{jj} = 1$ with other entries equal to zero. The term f_{ijk} is defined as follows:

$$f_{ijk} = \begin{cases} 1 & \text{if } Tr(\mathbf{M}^{ijk}\mathbf{U}\mathbf{U}^\top) > 0 \\ 0 & \text{otherwise} \end{cases}. \tag{5.9}$$

With models of independent and dependent information from distrust relations, we propose the following formulation to leverage distrust relations:

$$\min \|\mathbf{A}^n - \mathbf{U}\mathbf{H}^n\mathbf{U}^\top\|_F^2 + \sum_{\langle i,j,k\rangle \in \mathcal{S}} f_{ijk} Tr(\mathbf{M}^{ijk}\mathbf{U}\mathbf{U}^\top) \tag{5.10}$$

The Proposed Formulation for NCSSN

Combining Eqs. (5.4) and (5.10), the proposed node classification framework in trust/distrust networks NCSSN is to solve the following optimization problem:

$$\begin{aligned} \min_{\mathbf{H}^t \geq 0, \mathbf{H}^d \geq 0, \mathbf{U} \geq 0, \mathbf{W}} & \|\mathbf{T} - \mathbf{U}\mathbf{H}^t\mathbf{U}^\top\|_F^2 + \alpha\|\mathbf{C}(\mathbf{U}\mathbf{W} - \mathbf{Y})\|_F^2 \\ & + \beta\big(\|\mathbf{D} - \mathbf{U}\mathbf{H}^d\mathbf{U}^\top\|_F^2 + \sum_{\langle i,j,k\rangle \in \mathcal{S}} f_{ijk} Tr(\mathbf{M}^{ijk}\mathbf{U}\mathbf{U}^\top)\big) \\ & + \lambda(\|\mathbf{H}^t\|_F^2 + \|\mathbf{H}^d\|_F^2 + \|\mathbf{U}\|_F^2 + \|\mathbf{W}\|_F^2) \end{aligned} \tag{5.11}$$

where the first term to the fourth term capture trust information, label information from labeled nodes, independent information from distrust relations and dependent information from distrust relations, respectively. The term $\lambda(\|\mathbf{H}^t\|_F^2 + \|\mathbf{H}^d\|_F^2 + \|\mathbf{U}\|_F^2 + \|\mathbf{W}\|_F^2)$ is introduced to avoid overfitting. The relationships among $\mathbf{U}$, $\mathbf{H}^t$, $\mathbf{H}^d$ and $\mathbf{W}$ make the problem of finding optimal solutions for all parameters in Eq. (5.11) difficult to determine simultaneously. In this work, we adopt an alternate optimization scheme [16] for Eq. (5.11) where we optimize one component while fixing others. More details are shown in Appendix C. After we learn $\mathbf{W}$ and $\mathbf{U}$, the label of a node in the trust/distrust network $u_i \in \mathcal{U}^U$ can be predicted as follows:

$$c^* = \arg\max_{c_j \in \mathcal{C}}([\mathbf{U}_i\mathbf{W}]_j) \tag{5.12}$$

5.1.4 Evaluation

Dataset and Experimental Settings

Epinions can write reviews for products from various categories. We chose these categories as the class labels of users. For a user who writes reviews for products from multiple categories, we chose the one with most products she writes reviews to as her label. We perform additional preprocessing on the original Epinions dataset by filtering users without any labels, and class labels with a limited number of users. The new dataset includes 23,280 users, 291,422 trust relations, 40,792 distrust relations and 20 class labels.

In each case, we randomly choose $x\%$ of nodes labeled and the remaining $1-x\%$ as unlabeled nodes for testing. For each x, we repeat the experiments 10 times and report the average performance. Since it is very common for labels to be sparsely specified, we chose relatively small values of x, which were $\{5, 10, 15, 20\}$. One commonly used measure, referred to as Micro-F1, is adopted to assess the classification performance.

Node Classification Performance

In this subsection, we evaluate the classification performance with trust/distrust networks in terms of (a) the performance of algorithms transformed from trust networks, and (b) the performance of the proposed NCSSN framework with respect to these transformed methods.

The comparison results are demonstrated in Table 5.1. The algorithms in the table are defined as follows:

- *ICA*: This algorithm is a traditional iterative classification method [44] with features LF_{IP}, LF_{OP} and SF_P. We apply *ICA* to trust/distrust networks by ignoring all distrust relations.

- *sICA3*: This algorithm is a transformed version of *ICA* for trust/distrust networks. For SF_{PN}, we chose circles of length 3 as features.

- *sICA4*: This algorithm is also a transformed version of *ICA*. Different from *sICA3*, *sICA4* uses circles of length 4 as features.

- *GReg*: This algorithm is a traditional graph regularization method [105] for trust networks. We ignore all distrust relations when we apply *GReg* to trust/distrust networks.

- *disGReg*: This algorithm is a variant of *GReg*, which considers a distrust relation as dissimilarity.

- *sGReg*: This algorithm is also a variant of *GReg*. To apply *GReg* from trust to trust/distrust networks, *sGReg* converts distrust relations to trust relations according to status theory.

- *NCSSN*: This is the proposed algorithm which models both independent and dependent information to capture distrust relations.

- *Random*: this algorithm chooses class labels randomly for unlabeled nodes.

For methods with parameters, we used cross-validation to determine their values. However, *disGReg* achieves the best performance when $\eta = 0$. This means that the distrust relations in *disGReg* reduce its performance. To show the impact of distrust relations on *disGReg*, we report the performance when $\eta = 1$ in Table 5.1. Note that for *ICA*, *sICA3* and *sICA4*, we try various types of traditional classifiers including

Algorithms	Epinions			
	5%	10%	15%	20%
ICA	10.57	11.00	11.45	11.99
sICA3	11.09	11.61	12.03	12.52
sICA4	11.75	11.98	12.36	13.03
GReg	9.71	11.02	11.51	12.10
disGReg	9.39	10.70	11.21	11.80
sGReg	10.34	11.68	12.17	12.56
NCSSN	12.08	12.63	13.22	13.87
Random	4.98	5.00	5.00	5.02

Table 5.1: Performance Comparison of Negative Link Prediction in Epinions.

Naive Bayes, SVM, logistic regression, linear regression, and random forests, and we report the best performance.

We make some key observations from Table 5.1:

- In general, with the increase in the number of labeled nodes, the classification performance consistently increases for all methods in Table 5.1.

- The relative performance improvement of *sICA3* and *sICA4* compared to *ICA* is shown in Table 5.2. Both *sICA3* and *sICA4* outperform *ICA*. These results support the contention that distrust relations are helpful in the node classification problem. *sICA4* often outperforms *sICA3*. Since trust/distrust networks are usually very sparse, some users may not have any length 3 circle features but have 4-length circle features; hence length 4 circle features are more robust.

- The relative performance improvement of *disGReg* and *sGReg* compared to *GReg* is shown in Table 5.3. *disGReg* performs worse than *GReg*, which suggests

	sICA3	*sICA4*
5%	+4.92%	+11.16%
10%	+5.55%	+8.91%
15%	+5.07%	+7.95%
20%	+4.42%	+8.67%

Table 5.2: Relative Performance Improvement of *sICA3* and *sICA4* Compared to *ICA*.

	disGReg	*sGReg*
5%	-3.30%	+6.49%
10%	-2.90%	+5.73%
15%	-2.61%	+5.99%
20%	-2.48%	+3.80%

Table 5.3: Relative Performance Improvement of *disGReg* and *sGReg* Compared to *GReg*.

that a distrust relation may not denote dissimilarity and transforming *GReg* to trust/distrust by considering a distrust relation as dissimilarity may not work. These observations are consistent with our previous observations in Epinions that distrust in Epinions may not be dissimilarity measurements [83]. *sGReg* obtains better performance than *GReg*. Transforming *GReg* to trust/distrust networks by converting distrust relations to trust relations according to status theory can improve classification performance.

- The proposed framework *NCSSN* always obtains the best performance. *NCSSN* models independent and dependent information to capture distrust links and the contributions of these two components to the performance improvement from *NCSSN* will be discussed in the following subsection.

In summary, the aforementioned analysis provides the insights that (a) transformed algorithms from trust to trust/distrust networks properly can improve the classification performance; (b) a distrust relation may not denote dissimilarity; and (c) the proposed *NCSSN* framework obtains significant performance improvement compared to other methods.

Component Analysis for NCSSN

Based on the performance comparison in the previous subsection, we observe that the proposed *NCSSN* framework improves the classification performance significantly. To capture distrust relations, *NCSSN* provides two components $\|\mathbf{A}^n - \mathbf{U}\mathbf{H}^n\mathbf{U}^\top\|_F^2 + \sum_{\langle i,j,k\rangle \in \mathcal{S}} f_{ijk} Tr(\mathbf{M}^{ijk}\mathbf{U}\mathbf{U}^\top)$ to model independent and dependent information, respectively. To study the impact of distrust relations on the proposed framework, we systematically eliminate the effects of these two components by defining the following variants of *NCSSN*:

- NCSSN\II - We eliminate the effect of the independent component on *NCSSN*. In particular, we remove the term $\|\mathbf{A}^n - \mathbf{U}\mathbf{H}^n\mathbf{U}^\top\|_F^2$ from the optimization problem in Eq. (5.11).

- NCSSN\DI - We eliminate the effect of the dependent component on *NCSSN* by removing the term
  $\sum_{\langle i,j,k\rangle \in \mathcal{S}} f_{ijk} Tr(\mathbf{M}^{ijk}\mathbf{U}\mathbf{U}^\top)$ from the optimization problem in Eq. (5.11).

- NCSSN\DII - We eliminate the effects of both independent and dependent components on *NCSSN* by setting $\beta = 0$ in Eq. (5.11).

The parameters in all variants are determined with cross-validation and the performance comparison of *NCSSN* and its variants are demonstrated in Figure 5.3.

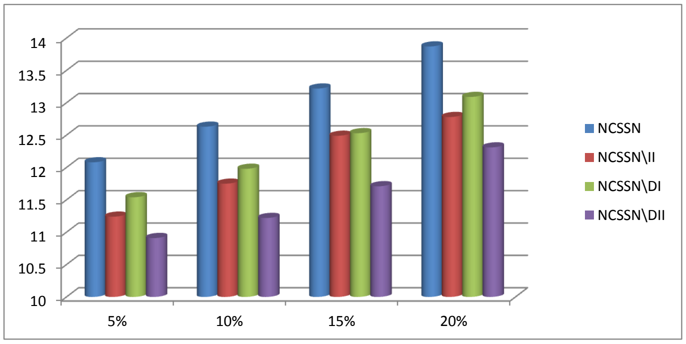


Figure 5.3: The Impact of Distrust Relations on The Proposed Framework.

When we eliminate the effect of the independent component, the performance of NCSSN\II degrades in comparison with *NCSSN*. For example, the performance reduces 6.95% with 5% of labeled users in Epinions. These results suggest that the independent component in *NCSSN* is important. We make similar observations for NCSSN\DI when we eliminate the effect of the component modeling dependent information. When we eliminate the effects of both components, the performance of NCSSN\DII further reduces compared to NCSSN\II and NCSSN\DI. These results suggest that these components are complementary to each other.

In summary, via the component analysis of *NCSSN*, we conclude that (a) both components can contribute to the performance improvement of *NCSSN*; (b) it is necessary to model both because they contain complementary information.

5.2 Recommendation

In the physical world, we always seek recommendations from our trusted friends, which suggests that trust information may be useful to improve recommendation performance. Many recommender systems are proposed to incorporate ones' trust networks for recommendation and gain performance improvement [52, 46, 27, 47, 28, 82, 91]. Scholars have noted that distrust information may be more noticeable and credible than trust information with a similar magnitude [12]. In this section, we investigate how to exploit trust/distrust networks for recommendation [72].

5.2.1 Problem Statement

In a typical recommender system, there is a user-item rating matrix $\mathbf{R} \in \mathbb{R}^{n \times I}$ where I is the number of the set of items $\mathcal{V} = \{v_1, v_2, \dots, v_I\}$, $\mathbf{R}_{ij}$ the rating score if u_i rates v_j, and 0 otherwise. Let $\mathcal{O} = \{\langle u_i, v_j \rangle | \mathbf{R}_{ij} \neq 0\}$ be the set of observed ratings and $\mathcal{M} = \{\langle u_i, v_k \rangle | \mathbf{R}_{ik} \neq 0\}$ be the set of missing ratings. The problem of recommendation with trust/distrust networks can be formally stated as follows:

Given observed ratings $\mathcal{O}$ and a trust/distrust network $\mathcal{G}$ with trust relations $\mathbf{T}$, and distrust relations $\mathbf{D}$, the problem of recommendation with a trust/distrust network aims to infer missing values $\mathcal{M}$ in $\mathbf{R}$.

5.2.2 A Recommendation Framework with Trust/Distrust Networks - RecSSN

Two types of information from trust networks can be exploited for recommendation, which correspond to local information and global information [87]. Local information reveals the correlations among the user and his/her trusted friends, while global information reveals the reputation of the user in the whole network. Users in the physical world are likely to ask for suggestions from their local friends while they also tend to seek suggestions from users with high global reputation. This suggests that both local and global information can be exploited in trust networks to improve the performance of recommender systems [81]. In the following subsections, we will first provide details about the methods for capturing local and global information in trust/distrust networks, and then introduce the proposed RecSSN framework.

Matrix factorization is chosen as our basic model because it is one of the most popular techniques for building recommender systems [35, 34]. Assume that $\mathbf{U}_i \in \mathbb{R}^K$ is the K-dimensional preference latent factor of u_i, and $\mathbf{V}_j \in \mathbb{R}^K$ is the K-dimensional characteristic latent factor of item j. Typically, scores from u_i to v_j in $\mathbf{R}_{ij}$ are modeled by the interactions between their latent factors. This interaction is defined in terms of the product of the latent vectors:

$$\mathbf{R}_{ij} = \mathbf{U}_i^\top \mathbf{V}_j \tag{5.13}$$

Matrix factorization-based recommender systems solve the following optimization problem:

$$\min \sum_{i=1}^{n} \sum_{j=1}^{m} \mathbf{W}_{ij} \|\mathbf{R}_{ij} - \mathbf{U}_i^\top \mathbf{V}_j\|_2^2 + \alpha(\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2) \tag{5.14}$$

where $\mathbf{U} = \{\mathbf{U}_1, \mathbf{U}_2, \dots, \mathbf{U}_n\}$ and $\mathbf{V} = \{\mathbf{V}_1, \mathbf{V}_2, \dots, \mathbf{V}_m\}$. $\mathbf{W}_{ij}$ controls the contribution from $\mathbf{R}_{ij}$, and the term $\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2$ is added to avoid overfitting.

Capturing Local Information from Trust/Distrust Networks

The local information in trust/distrust networks is about the preference relations between users, and their "friends" (or users with positive links) and "foes" (or users with negative links). Next, we introduce our approach to capture local information from trust/distrust networks based on the findings of the previous section.

Let $\mathcal{P}_i$ and $\mathcal{N}_i$ be u_i's friend circle, including users who have trust relations with u_i, and foe circle, including users who have distrust relations with u_i, respectively. Based on $\mathcal{P}_i$ and $\mathcal{N}_i$, we can divide users into three groups as below:

- $\mathcal{OP}$ includes users who have only trust links as - $\mathcal{OP} = \{u_i | \mathcal{P}_i \neq \emptyset \cap \mathcal{N}_i = \emptyset\}$;
- $\mathcal{ON}$ includes users who have only distrust links as - $\mathcal{ON} = \{u_i | \mathcal{P}_i = \emptyset \cap \mathcal{N}_i \neq \emptyset\}$;
- $\mathcal{PN}$ contains users who have both trust and distrust links as - $\mathcal{PN} = \{u_i | \mathcal{P}_i \neq \emptyset \cup \mathcal{N}_i \neq \emptyset\}$.

We define $\bar{U}_i^p$ and $\bar{U}_i^n$ as the average user preferences of u_i's friend circle and foe circle, respectively, as follows:

$$\bar{\mathbf{U}}_i^p = \frac{\sum_{u_j \in \mathcal{P}_i} \mathbf{S}_{ij} \mathbf{U}_j}{\sum_{u_j \in \mathcal{P}_i} \mathbf{S}_{ij}}, \quad \bar{\mathbf{U}}_i^n = \frac{\sum_{u_j \in \mathcal{N}_i} \mathbf{S}_{ij} \mathbf{U}_j}{\sum_{u_j \in \mathcal{N}_i} \mathbf{S}_{ij}} \tag{5.15}$$

where $\mathbf{S}_{ij}$ is the connection strength between u_i and u_j. Next, we will discuss how to capture local information for these groups separately:

- For a user u_i with only friend circle (or $u_i \in \mathcal{OP}$), our previous finding suggests that u_i's preference is likely to be similar with her friend circle. Hence, we force u_i's preference close to $\mathcal{P}_i$ by minimizing the following term:

$$\min \quad \|\mathbf{U}_i - \bar{\mathbf{U}}_i^p\|_2^2. \tag{5.16}$$

- For a user u_i with only foe circle (or $u_i \in \mathcal{ON}$), this user is likely to be untrustworthy and we should not consider this user for the purpose of recommendation [95]. Therefore, we ignore local information from these users with only foe circles, which are only a small portion of the users in real-world trust/distrust networks. For example, in the studied dataset, there are less than 5% of users with only foe circles.

- For a user u_i with both friend and foe circles, our previous finding suggests that the preference of u_i is likely to be closer to that of his/her friend circle than that of his/her foe circle. In other words, (1) if a user u_i sits closer to his/her friend circle $\mathcal{P}_i$ than her foe circle $\mathcal{N}_i$, i.e., $\|\mathbf{U}_i - \bar{\mathbf{U}}_i^p\|_2^2 - \|\mathbf{U}_i - \bar{\mathbf{U}}_i^n\|_2^2 < 0$, we should not penalize this case; while (2) if a user u_i sits closer to his/her foe circle $\mathcal{N}_i$ than her friend circle $\mathcal{P}_i$, i.e., $\|\mathbf{U}_i - \bar{\mathbf{U}}_i^p\|_2^2 - \|\mathbf{U}_i - \bar{\mathbf{U}}_i^n\|_2^2 > 0$, we should add a penalty to pull u_i closer to $\mathcal{P}_i$ than $\mathcal{N}_i$. Therefore, we propose the following minimization term to force u_i's preference closer to $\mathcal{P}_i$ than $\mathcal{N}_i$ as:

$$\min \quad \max(0, \|\mathbf{U}_i - \bar{\mathbf{U}}_i^p\|_2^2 - \|\mathbf{U}_i - \bar{\mathbf{U}}_i^n\|_2^2) \tag{5.17}$$

Next, we give details on the inner workings of Eq. (5.17). (1) When u_i sits closer to his/her friend circle $\mathcal{P}_i$ than his/her foe circle $\mathcal{N}_i$, the minimizing term in Eq. (5.17) is 0 because $\|\mathbf{U}_i - \bar{\mathbf{U}}_i^p\|_2^2 - \|\mathbf{U}_i - \bar{\mathbf{U}}_i^n\|_2^2 < 0$ and we do not add any penalty; and (2) when u_i sits closer to her foe circle $\mathcal{N}_i$ than her friend circle $\mathcal{P}_i$, the minimizing term in Eq. (5.17) is $\|\mathbf{U}_i - \bar{\mathbf{U}}_i^p\|_2^2 - \|\mathbf{U}_i - \bar{\mathbf{U}}_i^n$ because $\|\mathbf{U}_i - \bar{\mathbf{U}}_i^p\|_2^2 - \|\mathbf{U}_i - \bar{\mathbf{U}}_i^n > 0$ and Eq. (5.17) will pull u_i back to $\mathcal{P}_i$ from $\mathcal{N}_i$.

We can develop a unified term to capture local information from these three groups in trust/distrust networks with the following observations - (1) if we define $\bar{\mathbf{U}}_i^n = \mathbf{U}_i$ for u_i in $\mathcal{OP}$, the term for $\mathcal{OP}$ is equivalent to $\max(0, \|\mathbf{U}_i - \bar{\mathbf{U}}_i^p\|_2^2 - \|\mathbf{U}_i - \bar{\mathbf{U}}_i^n\|_2^2)$; and (2) if we define $\bar{\mathbf{U}}_i^n = \mathbf{U}_i$ for u_i in $\mathcal{ON}$, the term $\max(0, \|\mathbf{U}_i - \bar{\mathbf{U}}_i^p\|_2^2 - \|\mathbf{U}_i - \bar{\mathbf{U}}_i^n\|_2^2)$ is 0 for $\mathcal{ON}$, which indicates that we ignore the impact of users from $\mathcal{ON}$. Therefore by redefining $\bar{U}_i^p$ and $\bar{U}_i^n$ as,

$$\bar{\mathbf{U}}_i^p = \begin{cases} \frac{\sum_{u_j \in \mathcal{P}_i} \mathbf{S}_{ij}\mathbf{U}_j}{\sum_{u_j \in \mathcal{P}_i} \mathbf{S}_{ij}} & \text{for } u_i \in \mathcal{OP} \cup \mathcal{PN}, \\ \mathbf{U}_i & \text{for } u_i \in \mathcal{ON}. \end{cases}$$

$$\bar{\mathbf{U}}_i^n = \begin{cases} \frac{\sum_{u_j \in \mathcal{N}_i} \mathbf{S}_{ij}\mathbf{U}_j}{\sum_{u_j \in \mathcal{N}_i} \mathbf{S}_{ij}} & \text{for } u_i \in \mathcal{ON} \cup \mathcal{PN}, \\ \mathbf{U}_i & \text{for } u_i \in \mathcal{OP}, \end{cases} \tag{5.18}$$

we can find a unified term to capture local information from trust/distrust networks as:

$$\min \quad \sum_{i=1}^{n} \max(0, \|\mathbf{U}_i - \bar{\mathbf{U}}_i^p\|_2^2 - \|\mathbf{U}_i - \bar{\mathbf{U}}_i^n\|_2^2) \tag{5.19}$$

Capturing Global Information from Signed Social Networks

The global information of a trust/distrust network reveals the reputation of a user in the whole network [52]. User reputation is a sort of status that gives additional powers and capabilities in recommender systems [81]. There are many algorithms to calculate the reputations of nodes in trust networks [61, 32]. However, a small number of distrust links can significantly affect the status of the nodes, which suggests that we should consider distrust links. Therefore, we choose a variant of Pagerank, Exponential Ranking [93], taking into account distrust links to calculate user reputations. In detail, we first perform Exponential Ranking to rank users by exploiting the global information of trust/distrust networks. We assume that $r_i \in \{1, 2, \ldots, N\}$ is the reputation ranking of u_i where $r_i = 1$ denotes that u_i has the highest reputation in the trust network. Then we define user reputation score $\mathbf{w}_i$ as a function f of user reputation ranking r_i: $\mathbf{w}_i = f(r_i)$ where the function f limits the value of the reputation score w_i within $[0, 1]$ and is a decreasing function of r_i, i.e., top-ranked users have high reputation scores.

In the physical world, user reputation plays an important role in recommendation. Many companies employ people with high reputations to enhance consumers' awareness and understanding of their products. Seno and Lukas found that suggestions from people with high reputations positively affect a consumer's adoption of a brand [66]. While in the online world, Massa found that recommendations from users with high reputations are more likely to be trustworthy [52]. To capture global information from trust/distrust, we can use user reputation scores to weight the importance of their recommendations. Originally the importance of $\mathbf{R}_{ij}$ in Eq. (5.14) is controlled by $\mathbf{W}_{ij}$. With trust/distrust networks, we should also consider the reputation of u_i; hence we define the new weight for $\mathbf{R}_{ij}$ as $\hat{\mathbf{W}}_{ij} = g(\mathbf{W}_{ij}, \mathbf{w}_i)$ where g is a function to combine two weights. With these new weights, the formulation to capture global information from trust/distrust networks is computed as follows:

$$\min \sum_{i=1}^{N} \sum_{j=1}^{m} g(\mathbf{W}_{ij}, \mathbf{w}_i) \|\mathbf{R}_{ij} - \mathbf{U}_i^\top \mathbf{V}_j\|_2^2 + \alpha(\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2) \tag{5.20}$$

where the importance of $\mathbf{R}_{ij}$ is controlled by $\mathbf{W}_{ij}$ and the reputation score of u_i

through a function g.

An Optimization Algorithm for RecSSN

We have introduced our approaches to capture local and global information from trust/distrust networks. With these model components, we propose a recommendation framework, RecSSN, which exploits local and global information simultaneously from trust/distrust networks. The proposed RecSSN framework solves the following optimization problem:

$$\begin{aligned}\min \sum_{i=1}^{N}\sum_{j=1}^{m} g(\mathbf{W}_{ij}, \mathbf{w}_i)\|(\mathbf{R}_{ij} - \mathbf{U}_i\mathbf{V}_j^\top)\|_2^2 + \alpha(\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2) \\ + \beta\sum_{i=1}^{n} \max(0, \|\mathbf{U}_i - \bar{\mathbf{U}}_i^p\|_2^2 - \|\mathbf{U}_i - \bar{\mathbf{U}}_i^n\|_2^2)\end{aligned} \tag{5.21}$$

where $\beta\sum_{i=1}^{n} \max(0, \|\mathbf{U}_i - \bar{\mathbf{U}}_i^p\|_2^2 - \|\mathbf{U}_i - \bar{\mathbf{U}}_i^n\|_2^2)$ captures local information from trust/distrust networks and the parameter β controls its contribution. The term $g(\mathbf{W}_{ij}, \mathbf{w}_i)$ is introduced to capture global information from trust/distrust networks.

By setting $g(\mathbf{W}_{ij}, \mathbf{w}_i) = \mathbf{W}_{ij}$ and ignoring all distrust links, the proposed formulation for RecSSN in Eq. (5.21) can be written as follows:

$$\min \sum_{i=1}^{N}\sum_{j=1}^{m} \mathbf{W}_{ij}\|(\mathbf{R}_{ij} - \mathbf{U}_i\mathbf{V}_j^\top)\|_2^2 + \alpha(\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2) + \beta\sum_{i=1}^{n} \|\mathbf{U}_i - \bar{\mathbf{U}}_i^p\|_2^2 \tag{5.22}$$

Interestingly, this formulation is equivalent to one of the state-of-the-art recommender systems with trust networks SocialMF [27]. Therefore, RecSSN provides a unified recommendation framework with trust and trust/distrust networks. Eq. (5.21) is jointly convex with respect to $\mathbf{U}$ and $\mathbf{V}$ and there is no nice solution in closed form due to the use of the max function. A local minimum can be obtained through the gradient decent optimization method in Appendix D, which usually works well for recommender systems [35]. After learning the user preference matrix $\mathbf{U}$ and the item characteristic matrix $\mathbf{V}$, an unknown score $\hat{\mathbf{R}}_{i'j'}$ from the user u_i' to the item v_j' will be predicted as $\hat{\mathbf{R}}_{i'j'} = \mathbf{u}_{i'}^\top \mathbf{v}_{j'}$.

5.2.3 Evaluation

In this section, we conduct experiments to answer the following two questions - (1) can the proposed RecSSN framework improve the recommendation performance by exploiting trust/distrust networks? and (2) which model components of RecSSN contribute to the performance improvement? Before answering these questions, we begin by introducing the experimental settings.

Experimental Settings

Following common ways to assess recommendation performance in rating systems, we choose two metrics, corresponding to the Root Mean Square Error (RMSE) and the Mean Absolute Error (MAE), which are formally defined as follows:

$$\begin{aligned} RMSE &= \sqrt{\frac{\sum_{(u_i,v_j)\in\mathcal{T}} (\mathbf{R}_{ij} - \hat{\mathbf{R}}_{ij})^2}{|\mathcal{T}|}}, \\ MAE &= \frac{1}{|\mathcal{T}|} \sum_{(u_i,v_j)\in\mathcal{T}} |\mathbf{R}_{ij} - \hat{\mathbf{R}}_{ij}|, \end{aligned} \tag{5.23}$$

where $\mathcal{T}$ is the set of ratings in the testing set, $|\mathcal{T}|$ is the size of $\mathcal{T}$ and $\hat{\mathbf{R}}_{ij}$ is the predicted rating from u_i to v_j. A smaller RMSE or MAE value means better performance. Note that previous work demonstrated that *small improvement in RMSE or MAE terms can have a significant impact on the quality of the top few recommendations* [33]. In this work, we choose $x\%$ of rating scores as training and the remaining $1 - x\%$ as testing, and x is varied as $\{50, 70, 90\}$.

Performance Comparison of Recommender Systems

To answer the first question, we compare the proposed RecSSN framework with existing recommender systems. Traditional collaborative filtering systems can be grouped into memory-based systems and model-based systems; hence we choose two groups of baseline methods.

The first group of baseline methods includes the following memory-based systems:

- **UCF**: This system makes recommendations by aggregating recommendations from ones' similar users only based on the user-item matrix.
- **pUCF**: This system is a variant of **UCF**, which combines recommendations from ones' similar users and their trust friends [52]. **pUCF** utilizes both user-item matrix and trust links.
- **pnUCF**: This system is a variant of **pUCF**, which excludes recommendations from ones' foes by exploiting distrust links [95]. **pnUCF** makes use of user-item matrix, trust and distrust links.

The second group of baseline methods includes the following model-based systems:

- **MF**: This system performs matrix factorization on the user-item matrix as shown in Eq. (5.14) [64]. It only utilizes the user-item matrix.
- **SocialMF**: This system combines both user-item matrix and trust links for recommendation [27], which is a special case of the proposed framework with only trust links as shown in Eq. (5.22).
- **SoReg**: This system also leverages both user-item matrix and trust links, and defines social regularization to capture trust links [47].

- **LOCABAL**: This system captures local and global information of trust links under the matrix factorization framework [81].

- **disSoReg**: In [45], two systems are proposed to exploit trust and distrust links, respectively. **disSoReg** is a combination of these two systems to exploit trust and distrust links simultaneously, which is actually a variant of **SoReg** by considering distrust links as dissimilarity measurements.

Note that we use cross-validation to determine parameters for all baseline methods. For RecSSN, β is set to 0.7. We empirically set $\alpha = 0.1$ and the number of latent factors $K = 10$ for both datasets. In Eq. (5.20), we empirically find that $f(x) = \frac{1}{log(x+1)}$ and $g(x, y) = x * y$ work well. The comparison results are demonstrated in Tables 5.4.

We make the following observations:

- In general, model-based methods outperform memory-based methods on the two studied datasets. Most of the existing recommender systems suffer from the data sparsity problem but model-based methods are usually less sensitive than memory-based methods [33].

- **pUCF** outperforms **UCF**. Furthermore, **SocialMF**, **SoReg** and **LOCABAL** outperform **MF**. These results support the known contention that exploiting trust links can significantly improve recommendation performance.

- **LOCABAL** exploits local and global information from trust links, and obtains better performance than the systems which model only local information from trust links such as **SocialMF** and **SoReg**. These observations indicate the importance of global information for recommendation.

- **pnUCF** obtains better performance than **pUCF**, which suggests that excluding recommendations from users with distrust links can improve recommendation performance. Furthermore, **disSoReg** performs worse than **SoReg**. These results suggest that we may not consider distrust links as dissimilarities in recommendation, which is consistent with observations in [83].

- The proposed RecSSN framework always obtains the best performance. RecSSN captures local and global information from trust/distrust networks. In addition to trust links, trust/distrust networks also provide distrust links. More details about the effects of distrust links on the performance of RecSSN will be discussed in the following subsection.

With these observations, we can draw conclusions about the first question - the proposed RecSSN framework outperforms the state-of-the-art recommender systems by exploiting local and global information from trust/distrust networks.

Training	Metrics	Memory-based Methods			Model-based Methods					
		UCF	**pUCF**	**pnUCF**	**MF**	**SocialMF**	**SoReg**	**LOCABAL**	**disSoReg**	RecSSN
50%	MAE	1.0323	0.9764	0.9683	1.0243	0.9592	0.9589	0.9437	0.9679	0.9273
	RMSE	1.2005	1.1477	1.1392	1.1902	1.1397	1.1354	1.1212	1.1407	1.0886
70%	MAE	1.0074	0.9493	0.9402	0.9988	0.9341	0.9327	0.9274	0.9425	0.8981
	RMSE	1.1758	1.1301	1.1196	1.1692	1.1163	1.1127	1.1009	1.1237	1.0697
90%	MAE	0.9817	0.9272	0.9187	0.9779	0.9189	0.9153	0.9017	0.9263	0.8863
	RMSE	1.1592	1.1059	1.0885	1.1525	1.0986	1.0951	1.0821	1.1032	1.0479

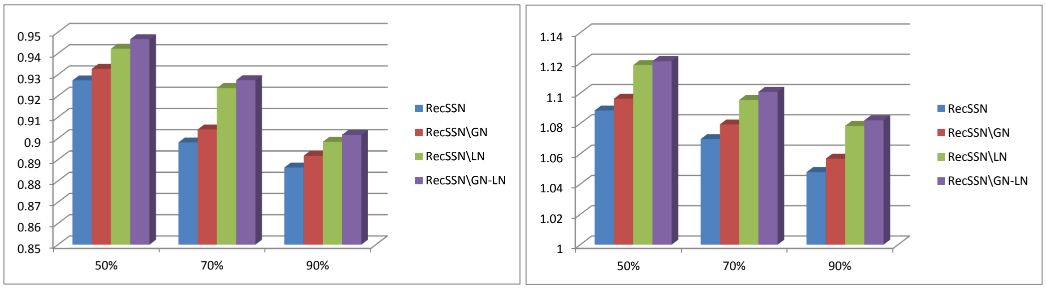
Table 5.4: Comparison of Different Recommender Systems in Epinions

Impact of Distrust Links on RecSSN

We will now focus on the second issue of examining the precise impact of distrust links on RecSSN. The experimental results in the previous subsection show that the proposed RecSSN framework outperforms various representative recommender systems with trust networks. Compared to these systems, RecSSN also leverages information from distrust links. In this subsection, we investigate the impact of distrust links on the proposed RecSSN framework to answer the second question. In particular, we eliminate the effects of distrust links systematically from RecSSN by defining the following algorithmic variants:

- *RecSSN\GN* - Eliminating the effect of distrust links from global information of trust/distrust networks by using Pagerank to calculate status scores of users with only trust links.
- *RecSSN\LN* - Eliminating the effect of distrust links from local information of trust/distrust networks by replacing $\sum_{i=1}^{n} \max(0, \|\mathbf{U}_i - \bar{\mathbf{U}}_i^p\|_2^2 - \|\mathbf{U}_i - \bar{\mathbf{U}}_i^n\|_2^2)$ with $\sum_{i=1}^{n} \|\mathbf{U}_i - \bar{\mathbf{U}}_i^p\|_2^2$ in Eq. (5.21).
- *RecSSN\GN-LN* - Eliminating the effects of distrust links from global and local information of trust/distrust networks.

The parameters in all these variants are determined via cross-validation. The experimental results are demonstrated in Figure 5.4. In general, eliminating any model component which captures the effect of distrust links will reduce the recommendation performance. The relative performance reductions for variants compared to RecSSN are shown in Table 5.5. When eliminating the effect of global information of distrust links from the proposed framework, the performance of *RecSSN\GN* degrades. We make a similar observation for *RecSSN\LN* when eliminating the effect of local in-



(a) AME

(b) RMSE

Figure 5.4: Impact of Distrust Links on The Proposed Framework RecSSN.

Variants	50%		70%		90%	
	MAE	RMSE	MAE	RMSE	MAE	RMSE
RecSSN\GN	-0.88%	-1.02%	-0.98%	-1.21%	-0.92%	-1.15%
RecSSN\LN	-2.06%	-3.06%	-3.15%	-2.71%	-1.67%	-3.21%
RecSSN\GN-LN	-2.59%	-3.29%	-3.56%	-3.22%	-2.04%	-3.56%

Table 5.5: Relative Performance Reductions for Variants Compared to RecSSN.

formation. For example, compared to RecSSN, *RecSSN\GN* and *RecSSN\LN* have 1.02% and 3.06% relative performance reductions, respectively, in terms of RMSE with 50% of Epinions data. When eliminating the effects of distrust links from global and local information of trust/distrust networks, *RecSSN\GN-LN* obtains worse performance than both *RecSSN\GN* and *RecSSN\LN*. This suggests that local and global information contain complementary information to each other for recommendation.

With the results from Figure 5.4 and Table 5.5, we can answer the second question - both local and global information of distrust links in the proposed RecSSN framework can help improve the recommendation performance.

5.3 Conclusion

In this chapter, we study two applications of distrust - node classification in trust/distrust networks and recommendation with trust/distrust networks. The research results suggest that (1) negative links may not denote dissimilarities; (2) the observation that users should site closer to their “friends” than their “foes” paves a way to capture trust/distrust relations; and (3) distrust have added value over trust and can significantly improve the performance of node classification and recommendation.

Chapter 6

GENERALIZING FINDINGS OF DISTRUST

Trust is a special type of positive links, and many properties and algorithms of trust can be generalized to positive links such as friendships and like [87]. For example, similar properties such as transitivity and homophily have been observed for positive links [99, 77]; trust prediction algorithms perform well with positive links [57, 77]; and many trust application frameworks can be directly applied to positive links [82, 46]. Since distrust is a special type of negative links, a natural question here is whether we can generalize some properties and algorithms of distrust to negative links such as foes and dislike. This investigation can greatly expand the boundaries of distrust computing and make research achievements be applicable to a wide range of applications.

We collect a dataset from Slashdot. Slashdot is a technology news platform in which users can create friend (positive) and foe (negative) links to other users. They can also post news articles. Other users may annotate these articles with their comments and opinions. Slashdot users can associate themselves with tags and join some interest groups. Some key statistics are demonstrated in Table 6.7. From the table, we note that Slashdot provides sufficient information to enable this investigation.

In the following sections, we will investigate whether we can generalize (1) properties, (2) distrust prediction algorithms and (3) application frameworks of distrust to negative links.

# of Users	7,275
# of Positive Links	67,705
# of Negative Links	20,851
# of Posts	300,932
# of Positive Opinions	1,742,763
# of Negative Opinions	42,260
# of Tags	27,942
# of Labels	10

Table 6.1: Statistics of the Slashdot Dataset.

6.1 Negative Link Properties

Distrust is not transitive, highly asymmetric and denotes neither similarity nor dissimilarity. In this section, we examine these properties of negative links. Note that since we use similar methods of distrust to investigate negative links, we omit details and directly present results and observations.

6.1.1 Transitivity

The results of transitivity of negative links are shown in Table 6.2. We make similar observations as - (1) positive links are transitive; (2) negative links are not transitive; and (3) when $\langle u_i\text{-}u_j,u_j\text{-}u_k\rangle$, it is also likely that u_i+u_k, which can be explained by balance theory as "enemies'enemies are friends".

6.1.2 Asymmetry

We show the results of asymmetry of negative links in Table 6.3. Positive links are asymmetric; while negative links are highly asymmetric. These observations are consistent with those for distrust.

Trust

Types	Number	P1	P2
$\langle u_i$+u_j,u_j+$u_k\rangle$, u_i?u_k	4,197,533	82.03%	N.A.
$\langle u_i$+u_j,u_j+$u_k\rangle$, u_i+u_k	898,905	17.57%	97.89%
$\langle u_i$+u_j,u_j+$u_k\rangle$, u_i-u_k	19,365	0.4%	2.11%

Distrust

Types	Number	P1	P2
$\langle u_i$-u_j,u_j-$u_k\rangle$, u_i?u_k	777,586	91.70%	N.A.
$\langle u_i$-u_j,u_j-$u_k\rangle$, u_i+u_k	13,362	1.67%	54.07%
$\langle u_i$-u_j,u_j-$u_k\rangle$, u_i-u_k	11,351	1.41%	45.93%

Table 6.2: Transitivity of Positive and Negative Links in Slashdot.

	u_j+u_i(%)	u_j-u_i(%)	u_j?u_i(%)
u_i+u_j	59,965(31.62)	556(0.29)	129,121(68.09)
u_i-u_j	556(1.69)	2,055(6.26)	30,218(92.05)

Table 6.3: Asymmetry of Positive and Negative Links.

6.1.3 Similarity

We investigate similarities between pairs with positive links, negative links and no links, and the average similarities are illustrated in Table 6.4. We observe - (1) pairs with positive links are likely to be similar; (2) pairs with negative links are more similar than those randomly selected pairs; and (3) pairs with positive links are more similar than those with negative links. Similar observations are made for distrust and trust relations.

	CI	COSINE	CI-COSINE
Foe ($\mathbf{s}_d$)	0.1926	0.1926	0.5577
Friend ($\mathbf{s}_t$)	0.2160	0.2160	0.6480
Random Pairs ($\mathbf{s}_r$)	0.1759	0.1759	0.5385

Table 6.4: Similarity for Positive and Negative Links.

6.2 Negative Link Prediction

We make a number of observations of distrust in Chapter 4 and based on these observations, we propose an unsupervised framework dTrust and a supervised framework NeLP to predict distrust by leveraging two sources - trust and content-centric user interactions. In this section, we first investigate whether we can make similar observations for negative links and then expand dTrust and NeLP to predict negative links by using positive links and content-centric user interactions.

6.2.1 Analysis on Negative Links

We compute the lengths of shortest paths of pairs with negative links in the positive networks and the length distribution is demonstrated in Figure 6.1. We make similar observations - more than 50% of our foes are within 2 hops and more than 80% of our foes are within 3 hops.

We examine all triads in Slashdot and find that (1) 93.01% of triads are balanced; and (2) 93.38% of triads satisfy status theory.

We investigate the existence of correlation between negative links and negative interactions via a two sample *t*-test. Evidence from *t*-test suggests that there is a strong correlation between negative links and negative interactions, and users with negative interactions are likely to have negative links. The distributions of ratios of negative links with respect to the number of negative interactions are shown in

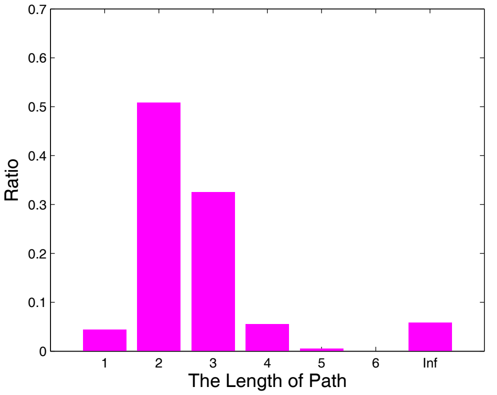


Figure 6.1: Ratio Distributions of the Length of Shortest Path for Pairs with Negative Links in the Positive Networks.

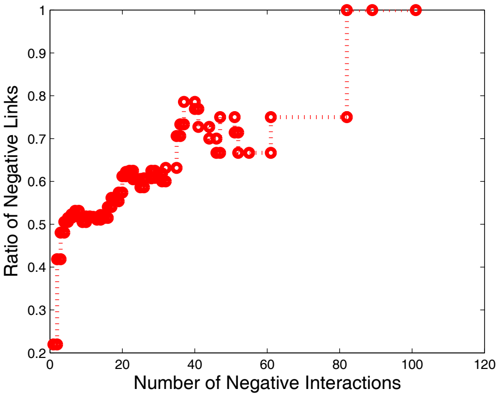


Figure 6.2: The Ratios of Negative Links with respect to the Number of Negative Interactions.

Figure 6.2. The ratio of a randomly selected pair as a negative link is $3.9402e - 04$ in Slashdot. Even when the numbers of negative interactions are small, the ratios are much higher than the random one, which further supports that existence of the correlation. Furthermore with increase of the number of negative interactions, the ratios tend to increase. Therefore, an increase in the number of negative interactions increases the likelihood of negative links between users.

Aforementioned analysis on negative links suggests that observations of distrust

	Slashdot50	Slashdot70	Slashdot100
negInter	0.0323	0.0819	0.0997
random	0.00001	0.00002	0.00003
dTrust	0.0754	0.1372	0.1491

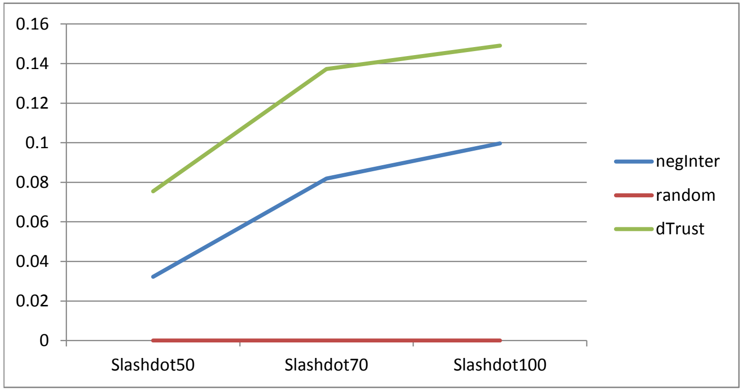


Figure 6.3: Performance of dTrust in Predicting Negative Links in Slashdot.

can be generalized to negative links, which indicates that the proposed frameworks dTrust and NeLP for distrust prediction may be generalized to predict negative links.

6.2.2 dTrust for Negative Link Prediction

Following similar experimental settings to assess dTrust in Epinions, we evaluate dTrust on Slashdot and the performance is illustrated in Figure 6.3. negInter obtains much better performance than random, which suggests the existence of the correlation between negative links and negative interactions. Performance comparison between dTrust and random suggests that dTrust can accurately predict negative links, which indicates that dTrust can be generalized to negative link prediction.

6.2.3 NeLP for Negative Link Prediction

We generalize NeLP to predict negative links by using positive links and content-centric user interactions in Slashdot and the performance is shown in Table 6.5. *negInS* always outperforms *negIn*, which indicates that status theory is helpful in

Algorithms	Slashdot	
	F1	Precision
random	0.0008	0.0004
sPath	0.0090	0.0172
negIn	0.1986	0.1483
negInS	0.2072	0.1524
NeLP-negIn	0.2394	0.2083
NeLP	0.2441	0.2139

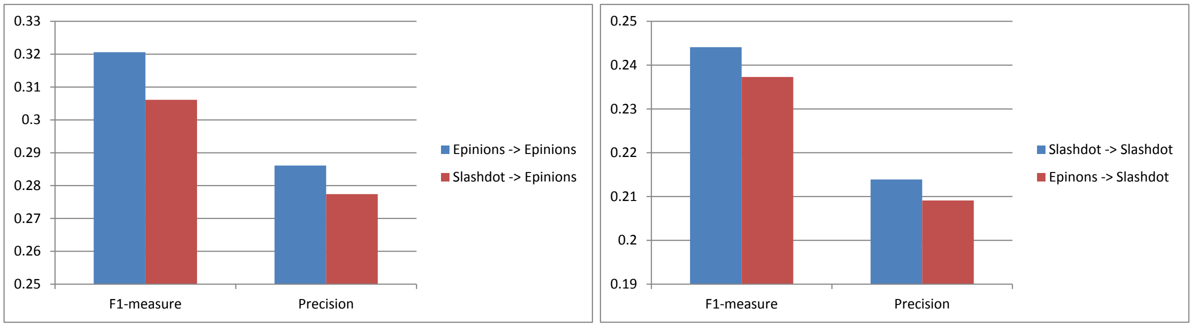
Table 6.5: Performance of NeLP in Predicting Negative Links in Slashdot.

negative link prediction. Compared to *random*, NeLP obtains much better performance, which suggests that NeLP can be generalized to negative link prediction.

Because the classifier learned by NeLP is based on the same set of features extracted from pervasively available sources for most social media sites, it is possible to generalize the classifier learned in one site to other sites and we further investigate how well the learned classifier generalizes across social media sites. In particular, we evaluate the performance of the classifier on Epinions (or Slashdot), which is learned from Slashdot (or Epinions). The results are shown in Figure 6.4. Note that in the figure $x \rightarrow y$ denotes training on x and evaluating on y. These results show that there is very good generalization of the classifier learned by NeLP although there is remarkably little decrease in performance regardless of which dataset is used for training.

6.3 Negative Link Applications

In Chapter 5, we propose a node classification framework NCSSN to infer labels of unlabeled nodes by leveraging labeled nodes and trust/distrust networks, and a rec-



(a) Epinions

(b) Slashdot

Figure 6.4: The Negative Link Prediction Performance across Epinions and Slashdot.

ommendation framework RecSSN, which exploits trust/distrust networks to improve recommendation performance. In this section, we investigate the generalization of NCSSN and RecSSN from trust/distrust networks to signed social networks (or networks with positive and negative links).

6.3.1 NCSSN for Node Classification in Signed Social Networks

Similar to evaluation of NCSSN in Epinions, we assess NCSSN in Slashdot. Users in Slashdot can join in some interest groups and these group identifiers are treated as the class labels. Via cross validation, the parameters for *NCSSN* are set as $\{\alpha = 1, \beta = 0.5, K = 500\}$ and the experimental results are demonstrated in Table 6.6. We observe that - (1) *sICA3* and *sICA4* outperform *ICA* and negative links can improve node classification performance; (2) *disGReg* performs worse than *GReg*, and negative links may not denote dissimilarities; and (3) performance comparison between *NCSSN* and *Random* indicates that *NCSSN* can significantly improve node classification performance with signed social networks. All these observations suggest that *NCSSN* can be generalized to the problem of node classification in signed social

Algorithms	Slashdot			
	5%	10%	15%	20%
ICA	19.99	20.86	21.25	21.34
sICA3	21.46	21.68	22.31	22.75
sICA4	22.24	22.30	22.85	23.13
GReg	19.57	20.91	21.23	22.03
disGReg	18.93	19.85	20.54	21.19
sGReg	20.56	22.56	22.77	22.90
NCSSN	23.54	24.66	25.20	25.62
Random	8.33	8.31	8.34	8.34

Table 6.6: Performance Comparison of Node Classification in Slashdot.

networks.

6.3.2 RecSSN for Recommendation with Signed Social Networks

In Slashdot, users are associated with certain tags and the recommendation task is to recommend tags to users. In this scenario, the performance is often evaluated via precision@N and recall@N [67], which are formally defined as follows:

$$precision@N = \frac{\sum_{u_i \in \mathcal{U}} |TopN_i \bigcap I_i|}{\sum_{u_i \in \mathcal{U}} |TopN_i|} \tag{6.1}$$

$$recall@N = \frac{\sum_{u_i \in \mathcal{U}} |TopN_i \bigcap I_i|}{\sum_{u_i \in \mathcal{U}} |I_i|}, \tag{6.2}$$

where $TopN_i$ is the set of N items recommended to user u_i that u_i has not been associated in the training set, and I_i is the set of items that have been associated with u_i in the testing set. A larger precision@N or recall@N value means better performance. The values of precision@N and recall@N are usually small in the case of sparse datasets. For example, the precision@5 is less than 0.05 over a dataset with

Metrics	Memory-based Methods			Model-based Methods					
	UCF	**pUCF**	**pnUCF**	**MF**	**SocialMF**	**SoReg**	**LOCABAL**	**disSoReg**	RecSSN
P@5	0.0343	0.0372	0.0381	0.0354	0.0387	0.0386	0.0394	0.0379	0.0419
R@5	0.0438	0.0479	0.0485	0.0453	0.0492	0.0488	0.0498	0.0473	0.0511
P@10	0.0332	0.0358	0.0364	0.0338	0.0365	0.0368	0.0375	0.0359	0.0388
R@10	0.0413	0.0454	0.0463	0.0427	0.0463	0.0467	0.0479	0.0457	0.0497

Table 6.7: Comparison of Different Recommender Systems in Slashdot.

$8.02e-3$ density [17, 18]. In this work, we set $N=5$ and $N=10$.

The recommendation performance is illustrated in Figure 6.7. Note that we set $\beta=0.3$ via cross validation. It is observed that (1) **disSoReg** obtains worse performance than **SoReg** and we may not consider negative links as dissimilarities in recommendation; and (2) RecSSN can significantly improve recommendation performance by exploiting signed social networks compared to baseline methods. These observations suggest that the recommendation framework RecSSN can be generalized from trust/distrust networks to signed social networks.

6.4 Conclusion

Some properties and algorithms of trust can be generalized to positive links, which motivates us to study whether we can generalize findings of distrust to negative links. The aforementioned investigations suggest that (1) similar properties are observed for negative links as distrust; (2) distrust prediction frameworks dTrust and NeLP can accurately predict negative links; and (3) application frameworks NCSSN and RecSSN can be generalized from trust/distrust networks to signed social networks (or networks with positive and negative links).

Chapter 7

CONCLUSION AND FUTURE WORK

In this chapter, we summarize our research results and their broader impacts, and discuss promising research directions.

7.1 Summary

In this dissertation, we propose four innovative research tasks - (1) understanding distrust; (2) predicting distrust; (3) applying distrust and (4) generalizing findings of distrust.

For understanding distrust, we investigate properties of distrust and find that we can not extend properties of trust to distrust and distrust presents distinct properties. The computational task of predicting distrust from only trust suggests that we can not predict distrust from only trust hence distrust is not the negation of trust in social media; while the computational task of predicting trust with information from distrust indicates that distrust can significantly improve trust prediction performance hence distrust has added value over trust.

For predicting distrust, we formally define the problem and make a number of important findings about distrust - (1) our "foes" are close to us in the trust network; (2) most triads satisfy balance and status theories; (3) there is a strong correlation between distrust and negative interactions; and (4) negative interactions between users increase the propensity of distrust. These findings serve as the groundwork of an unsupervised framework dTrust and a supervised framework NeLP, which can predict distrust accurately by leveraging trust and content-centric user interactions.

For applying distrust, we propose principled approaches to model distrust in two representative social media applications, i.e., node classification and recommendation. The successful experiences of applying distrust in node classification and recommendation suggest that (1) distrust may not denote dissimilarity although trust denotes similarity; and (2) distrust has potentials in improving the performance of social media applications.

For generalizing findings of distrust, we investigate the generalization of properties and algorithms of distrust to negative links. We find that (1) negative links show similar properties as distrust; (2) distrust predication frameworks dTrust and NeLP can accurately predict negative links; and (3) the node classification framework NC-SSN and the recommendation framework RecSSN can be successfully expanded for negative links.

This dissertation investigates original problems that entreat unconventional data mining solutions. They are challenging because distrust is often not available in social media and they are original because little is known about distrust and its role in social media applications. Methodologies and techniques presented in this dissertation also have broader impacts:

- Data availability is still a challenging problem for social scientists [98]. Social media provides a virtual world for users online activities and makes it possible for social scientists to observe social behavior and interaction data of hundreds of millions of users. Our successful experiences of using social media data to study the social concept distrust pave the way for new research endeavors to enable the large-scale study of user behaviors in social media in computational social science.

- The enabling of distrust in social media and the successfully applying distrust in social media applications not only can have impact on industrial IT applications by improving services and user experience but also will open doors to new opportunities of research and development involving social media.

- Social theories are useful to explain user behaviors in social media and play important roles in helping distrust computing in social media. The techniques of modeling social theories may be directly applied to social media mining tasks [102] and their success manifests a new research direction - mining social media data with social theories.

7.2 Future Work

Computing distrust in social media is still in its early stages of development and an active area of exploration. Below we present some promising research directions:

- **Distrust Prediction with Cross-media Data:** Our previous study suggests that the learned distrust predictor by NeLP have very good generalization across social media sites, which suggests not only that some underlying general principles guide the creation of distrust relations but also that cross-media data has potentials in distrust prediction. The key issue of the problem of distrust prediction with cross-media data is how to transfer knowledge or patterns from distrust in the source site to the target site. This would be an application scenario of transfer learning [62]; hence, we will investigate under what circumstances, transfer learning algorithms are applicable for distrust prediction with cross-media data.

- **Evaluation without Ground Truth:** Evaluations in this dissertation are based on datasets with ground truth. However distrust is usually unavailable in social media and then the chief challenge of evaluation is related to the invisibility of distrust - how to verify the correctness of predicted distrust when they are invisible. We can take a multipronged approach to evaluation challenges. Traditionally, *training and test* datasets are used in evaluation; in such as a case, we say there is ground truth. When these datasets are not available, *user studies* are conducted (e.g., employing Amazon Mechanical Turks). We also can try user studies as in a study [100] to verify the performance of predicting strong and weak ties on Facebook. In essence, this method would rely on a group of recruited subjects who donate their data but withhold distrust and then compare predicted distrust against their withheld distrust.

- **Putting Distrust into More Social Media Applications:** We use node classification and recommendation as examples to illustrate that distrust can significantly improve their performance. The exciting progress not only proves the importance of distrust but also suggests that we should put distrust into more social media applications. The enabling and generalizing of distrust further broaden its applications. We will investigate how to apply distrust in more social media applications such as data clustering, active learning, information propagation, sentiment analysis and feature selection [74].

REFERENCES

[1] M. A. Abbasi, J. Tang, and H. Liu. Scalable learning of users preferences using networked data. In *Proceedings of the 25th ACM conference on Hypertext and social media*, pages 4–12. ACM, 2014.

[2] M. A. Abbasi, J. Tang, and H. Liu. Trust-aware recommender systems. *Machine Learning book on computational trust,Chapman & Hall/CRC Press*, 2014.

[3] B. Barber. *The logic and limits of trust.* Rutgers University Press New Brunswick, NJ, 1983.

[4] M. Belkin, P. Niyogi, and V. Sindhwani. On manifold regularization. In *Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics (AISTAT 2005)*, pages 17–24. Citeseer, 2005.

[5] S. Bhagat, G. Cormode, and S. Muthukrishnan. Node classification in social networks. In *Social network data analytics*, pages 115–148. Springer, 2011.

[6] A. Blum and S. Chawla. Learning from labeled and unlabeled data using graph mincuts. In *Proceedings of the Eighteenth International Conference on Machine Learning*, pages 19–26. Morgan Kaufmann Publishers Inc., 2001.

[7] G. Cai, J. Tang, and Y. Wen. Trust prediction with temporal dynamics. In *Web-Age Information Management*, 2014.

[8] D. Cartwright and F. Harary. Structural balance: a generalization of heider's theory. *Psychological Review*, 63(5):277, 1956.

[9] Y. Chang, L. Tang, Y. Inagaki, and Y. Liu. What is tumblr: A statistical overview and comparison. *ACM SIGKDD Explorations Newsletter*, 16(1):21–29, 2014.

[10] K.-Y. Chiang, C.-J. Hsieh, N. Natarajan, A. Tewari, and I. S. Dhillon. Prediction and clustering in signed networks: A local to global perspective. *arXiv preprint arXiv:1302.5145*, 2013.

[11] K.-Y. Chiang, N. Natarajan, A. Tewari, and I. S. Dhillon. Exploiting longer cycles for link prediction in signed networks. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 1157–1162. ACM, 2011.

[12] J. Cho. The mechanism of trust and distrust formation and their relational outcomes. *Journal of retailing*, 82(1):25–35, 2006.

[13] P. Cofta. Distrust. In *ICEC*. ACM, 2006.

[14] N. Cristianini and J. Shawe-Taylor. *An introduction to support vector machines and other kernel-based learning methods.* Cambridge university press, 2000.

[15] M. Cygan, M. Pilipczuk, M. Pilipczuk, and J. O. Wojtaszczyk. Sitting closer to friends than enemies, revisited. In *Mathematical Foundations of Computer Science 2012*, pages 296–307. Springer, 2012.

[16] C. Ding, T. Li, W. Peng, and H. Park. Orthogonal nonnegative matrix t-factorizations for clustering. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 126–135. ACM, 2006.

[17] H. Gao, J. Tang, X. Hu, and H. Liu. Exploring temporal effects for location recommendation on location-based social networks. In *Proceedings of the 7th ACM conference on Recommender systems*, pages 93–100. ACM, 2013.

[18] H. Gao, J. Tang, X. Hu, and H. Liu. Content-aware point of interest recommendation on location-based social networks. In *AAAI*. AAAI, 2015.

[19] L. Getoor and C. P. Diehl. Link mining: a survey. *ACM SIGKDD Explorations Newsletter*, 7(2):3–12, 2005.

[20] J. Golbeck. Computing and applying trust in web-based social networks. *Ph.D. dissertation*, 2005.

[21] J. Golbeck. Generating predictive movie recommendations from trust in social networks. *Trust Management*, pages 93–104, 2006.

[22] J. Golbeck. *Computing with social trust*. Springer Publishing Company, Incorporated, 2008.

[23] R. Guha, R. Kumar, P. Raghavan, and A. Tomkins. Propagation of trust and distrust. In *Proceedings of the 13th international conference on World Wide Web*, pages 403–412. ACM, 2004.

[24] R. Hardin. *Distrust: Manifestations and management*. Russell Sage Foundation, 2004.

[25] F. Heider. Attitudes and cognitive organization. *The Journal of psychology*, 21(1):107–112, 1946.

[26] X. Hu, L. Tang, J. Tang, and H. Liu. Exploiting social relations for sentiment analysis in microblogging. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pages 537–546. ACM, 2013.

[27] M. Jamali and M. Ester. Trustwalker: a random walk model for combining trust-based and item-based recommendation. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 397–406. ACM, 2009.

[28] M. Jiang, P. Cui, R. Liu, Q. Yang, F. Wang, W. Zhu, and S. Yang. Social contextual recommendation. In *Proceedings of the 22th ACM international conference on Information and knowledge management*. ACM, 2012.

[29] A. Josang, E. Gray, and M. Kinateder. Analysing topologies of transitive trust. In *Proc. of the 1st workshop on Formal Aspects in Security and Trust (FAST2003)*, 2003.

[30] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina. The eigentrust algorithm for reputation management in p2p networks. In *Proceedings of the 12th international conference on World Wide Web*, pages 640–651. ACM, 2003.

[31] D. Kempe, J. Kleinberg, and É. Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 137–146. ACM, 2003.

[32] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, 46(5):604–632, 1999.

[33] Y. Koren. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 426–434. ACM, 2008.

[34] Y. Koren. Collaborative filtering with temporal dynamics. *Communications of the ACM*, 53(4):89–97, 2010.

[35] Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

[36] R. M. Kramer. Trust and distrust in organizations: Emerging perspectives, enduring questions. *Annual review of psychology*, 50(1):569–598, 1999.

[37] D. W. Larson and R. Hardin. Distrust: Prudent, if not always wise. *Distrust*, pages 34–59, 2004.

[38] J. Leskovec, D. Huttenlocher, and J. Kleinberg. Predicting positive and negative links in online social networks. In *Proceedings of the 19th international conference on World wide web*, pages 641–650. ACM, 2010.

[39] J. Leskovec, D. Huttenlocher, and J. Kleinberg. Signed networks in social media. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1361–1370. ACM, 2010.

[40] R. J. Lewicki, D. J. McAllister, and R. J. Bies. Trust and distrust: New relationships and realities. *Academy of management Review*, 23(3):438–458, 1998.

[41] D. Liben-Nowell and J. Kleinberg. The link-prediction problem for social networks. *Journal of the American society for information science and technology*, 58(7):1019–1031, 2007.

[42] R. N. Lichtenwalter, J. T. Lussier, and N. V. Chawla. New perspectives and methods in link prediction. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 243–252. ACM, 2010.

[43] H. Liu, E. Lim, H. Lauw, M. Le, A. Sun, J. Srivastava, and Y. Kim. Predicting trusts among users of online communities: an epinions case study. In *Proceedings of the 9th ACM Conference on Electronic Commerce*, pages 310–319. ACM, 2008.

[44] Q. Lu and L. Getoor. Link-based classification. In *ICML*, volume 3, pages 496–503, 2003.

[45] H. Ma, M. R. Lyu, and I. King. Learning to recommend with trust and distrust relationships. In *Proceedings of the third ACM conference on Recommender systems*, pages 189–196. ACM, 2009.

[46] H. Ma, H. Yang, M. Lyu, and I. King. Sorec: social recommendation using probabilistic matrix factorization. In *Proceeding of the 17th ACM conference on Information and knowledge management*, pages 931–940. ACM, 2008.

[47] H. Ma, D. Zhou, C. Liu, M. Lyu, and I. King. Recommender systems with social regularization. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 287–296. ACM, 2011.

[48] N. Ma, E. Lim, V. Nguyen, A. Sun, and H. Liu. Trust relationship prediction using online product review data. In *Proceeding of the 1st ACM international workshop on Complex networks meet information & knowledge management*, pages 47–54. ACM, 2009.

[49] S. A. Macskassy and F. Provost. A simple relational classifier. Technical report, DTIC Document, 2003.

[50] P. V. Marsden and N. E. Friedkin. Network studies of social influence. *Sociological Methods & Research*, 22(1):127–151, 1993.

[51] S. Marsh and M. R. Dibben. Trust, untrust, distrust and mistrust–an exploration of the dark (er) side. In *Trust Management*, pages 17–33. Springer, 2005.

[52] P. Massa. A survey of trust use and modeling in real online systems. *Trust in E-services: Technologies, Practices and Challenges*, 2007.

[53] P. Massa and P. Avesani. Trust-aware recommender systems. In *Proceedings of the 2007 ACM conference on Recommender systems*, pages 17–24. ACM, 2007.

[54] D. H. McKnight and N. L. Chervany. Trust and distrust definitions: One bite at a time. In *Trust in Cyber-societies*, pages 27–54. Springer, 2001.

[55] D. H. McKnight and V. Choudhury. Distrust and trust in b2c e-commerce: Do they differ? In *ICEC*, pages 482–491. ACM, 2006.

[56] M. McPherson, L. Smith-Lovin, and J. M. Cook. Birds of a feather: Homophily in social networks. *Annual review of sociology*, pages 415–444, 2001.

[57] A. K. Menon and C. Elkan. Link prediction via matrix factorization. In *Machine Learning and Knowledge Discovery in Databases*, pages 437–452. Springer, 2011.

[58] J. Neville and D. Jensen. Iterative classification in relational data. In *Proc. AAAI-2000 Workshop on Learning Statistical Models from Relational Data*, pages 13–20, 2000.

[59] V. Nguyen, E. Lim, J. Jiang, and A. Sun. To trust or not to trust? predicting online trusts using trust antecedent framework. In *Ninth IEEE International Conference on Data Mining*, pages 896–901. IEEE, 2009.

[60] J. Nocedal and S. Wright. *Numerical optimization*. Springer verlag, 1999.

[61] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. 1999.

[62] S. J. Pan and Q. Yang. A survey on transfer learning. *Knowledge and Data Engineering, IEEE Transactions on*, 22(10):1345–1359, 2010.

[63] J. B. Rotter. Interpersonal trust, trustworthiness, and gullibility. *American psychologist*, 35(1):1, 1980.

[64] R. Salakhutdinov and A. Mnih. Probabilistic matrix factorization. *Advances in neural information processing systems*, 20:1257–1264, 2008.

[65] P. Sen, G. Namata, M. Bilgic, L. Getoor, B. Galligher, and T. Eliassi-Rad. Collective classification in network data. *AI magazine*, 29(3):93, 2008.

[66] D. Seno and B. Lukas. The equity effect of product endorsement by celebrities: A conceptual framework from a co-branding perspective. *European Journal of Marketing*, 2007.

[67] B. Sigurbjörnsson and R. Van Zwol. Flickr tag recommendation based on collective knowledge. In *Proceedings of the 17th international conference on World Wide Web*, pages 327–336. ACM, 2008.

[68] J. Singh and D. Sirdeshmukh. Agency and trust mechanisms in consumer satisfaction and loyalty judgments. *Journal of the Academy of Marketing Science*, 28(1):150–167, 2000.

[69] M. Szell, R. Lambiotte, and S. Thurner. Multirelational organization of large-scale social networks in an online world. *Proceedings of the National Academy of Sciences*, 107(31):13636–13641, 2010.

[70] C. Tan, L. Lee, J. Tang, L. Jiang, M. Zhou, and P. Li. User-level sentiment analysis incorporating social networks. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1397–1405. ACM, 2011.

[71] J. Tang, C. Aggarwal, and H. Liu. Node classification in signed social networks. In *Submitted to ACM SIGKDD international conference on Knowledge discovery and data mining*, 2015.

[72] J. Tang, C. Aggarwal, and H. Liu. Recommendation with signed social networks. In *Submitted to The 37th Annual ACM SIGIR conference*, 2015.

[73] J. Tang, S. Chang, C. Aggarwal, and H. Liu. Negative link prediction in social media. In *ACM International Conference on Web Search and Data Mining*, 2015.

[74] J. Tang, Y. Chang, C. Aggarwal, and H. Liu. A survey of mining signed networks in social media. *Submitted to ACM Computing Survey*, 2015.

[75] J. Tang, H. Gao, A. Dassarma, Y. Bi, and H. Liu. Trust evolution: Modeling and its applications. *IEEE Transactions on Knowledge and Data Engineering*, 2013.

[76] J. Tang, H. Gao, X. Hu, and H. Liu. Context-aware review helpfulness rating prediction. In *Proceedings of the 7th ACM conference on Recommender systems*, pages 1–8. ACM, 2013.

[77] J. Tang, H. Gao, X. Hu, and H. Liu. Exploiting homophily effect for trust prediction. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pages 53–62. ACM, 2013.

[78] J. Tang, H. Gao, and H. Liu. mTrust: Discerning multi-faceted trust in a connected world. In *the 5th ACM International Conference on Web Search and Data Mining*, 2012.

[79] J. Tang, H. Gao, H. Liu, and A. Das Sarma. eTrust: Understanding trust evolution in an online world. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 253–261. ACM, 2012.

[80] J. Tang, X. Hu, Y. Chang, and H. Liu. Predictability of distrust with interaction data. In *ACM International Conference on Information and Knowledge Management*, 2014.

[81] J. Tang, X. Hu, H. Gao, and H. Liu. Exploiting local and global social context for recommendation. In *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, pages 2712–2718. AAAI Press, 2013.

[82] J. Tang, X. Hu, and H. Liu. Social recommendation: a review. *Social Network Analysis and Mining*, 3(4):1113–1133, 2013.

[83] J. Tang, X. Hu, and H. Liu. Is distrust the negation of trust? the value of distrust in social media. In *ACM Hypertext conference*, 2014.

[84] J. Tang and H. Liu. Feature selection with linked data in social media. In *SDM*, pages 118–128. SIAM, 2012.

[85] J. Tang and H. Liu. Unsupervised feature selection for linked social media data. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 904–912. ACM, 2012.

[86] J. Tang and H. Liu. Feature selection for social media data. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 8(4):19, 2014.

[87] J. Tang and H. Liu. Trust in social computing. In *Proceedings of the companion publication of the 23rd international conference on World wide web companion*, pages 207–208. International World Wide Web Conferences Steering Committee, 2014.

[88] J. Tang and H. Liu. An unsupervised feature selection framework for social media data. *IEEE Transactions on Knowledge and Data Engineering*, 2014.

[89] J. Tang and H. Liu. *Trust in Social Media.* Morgan & Claypool Publishers, 2015.

[90] J. Tang, C. Nobata, A. Dong, Y. Chang, and H. Liu. Propagation-based sentiment analysis for microblogging data. In *SIAM International Conference on Data Mining*, 2015.

[91] J. Tang, J. Tang, and H. Liu. Recommendation in social media: recent advances and new frontiers. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1977–1977. ACM, 2014.

[92] B. Taskar, P. Abbeel, and D. Koller. Discriminative probabilistic models for relational data. In *Proceedings of the Eighteenth conference on Uncertainty in artificial intelligence*, pages 485–492. Morgan Kaufmann Publishers Inc., 2002.

[93] V. Traag, Y. Nesterov, and P. Van Dooren. Exponential ranking: Taking into account negative links. *Social Informatics*, pages 192–202, 2010.

[94] P. Victor, C. Cornelis, M. De Cock, and P. P. Da Silva. Gradual trust and distrust in recommender systems. *Fuzzy Sets and Systems*, 160(10):1367–1382, 2009.

[95] P. Victor, C. Cornelis, M. De Cock, and A. Teredesai. Trust-and distrust-based recommendations for controversial reviews. In *Web Science Conference (WebSci'09: Society On-Line)*, number 161, 2009.

[96] F. Wang, T. Li, X. Wang, S. Zhu, and C. Ding. Community discovery using nonnegative matrix factorization. *Data Mining and Knowledge Discovery*, 22(3):493–521, 2011.

[97] Y. Wang, X. Wang, J. Tang, W. Zuo, and G. Cai. Modeling status theory in trust prediction. In *the AAAI Conference on Artificial Intelligence*, 2015.

[98] D. J. Watts. Computational social science: Exciting progress and future directions. *The Bridge on Frontiers of Engineering*, 43(4):5–10, 2013.

[99] J. Weng, E. Lim, J. Jiang, and Q. He. Twitterrank: finding topic-sensitive influential twitterers. In *Proceedings of the third ACM international conference on Web search and data mining*, pages 261–270. ACM, 2010.

[100] R. Xiang, J. Neville, and M. Rogati. Modeling relationship strength in online social networks. In *Proceedings of the 19th international conference on World wide web*, pages 981–990. ACM, 2010.

[101] S.-H. Yang, A. J. Smola, B. Long, H. Zha, and Y. Chang. Friend or frenemy?: predicting signed ties in social networks. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, pages 555–564. ACM, 2012.

[102] R. Zafarani, M. A. Abbasi, and H. Liu. *Social Media Mining: An Introduction.* Cambridge University Press, 2014.

[103] E. Zheleva and L. Getoor. To join or not to join: the illusion of privacy in social networks with mixed public and private user profiles. In *Proceedings of the 18th international conference on World wide web*, pages 531–540. ACM, 2009.

[104] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf. Learning with local and global consistency. *Advances in neural information processing systems*, 16(16):321–328, 2004.

[105] D. Zhou, J. Huang, and B. Schölkopf. Learning from labeled and unlabeled data on a directed graph. In *Proceedings of the 22nd international conference on Machine learning*, pages 1036–1043. ACM, 2005.

[106] X. Zhu, Z. Ghahramani, J. Lafferty, et al. Semi-supervised learning using gaussian fields and harmonic functions. In *ICML*, volume 3, pages 912–919, 2003.

[107] C. Ziegler and J. Golbeck. Investigating interactions of trust and interest similarity. *Decision Support Systems*, 43(2):460–475, 2007.

[108] C.-N. Ziegler and G. Lausen. Propagation models for trust and distrust in social networks. *Information Systems Frontiers*, 7(4-5):337–358, 2005.

APPENDIX A

AN OPTIMIZATION ALGORITHM FOR DISTRUST

Set $\mathbf{A} = \mathbf{F} - \lambda \mathbf{r}\mathbf{r}^\top$ and let $\mathcal{L}$ contain terms related to $\mathbf{U}$ and $\mathbf{H}$ in the objective function $\mathcal{J}$ of Eq. (4.9), which can be rewritten as,

$$\begin{aligned}\mathcal{L} = Tr(&-2(\mathbf{W} \odot \mathbf{W} \odot \mathbf{A})\mathbf{U}\mathbf{H}^\top\mathbf{U}^\top + (\mathbf{W} \odot \mathbf{W} \odot \mathbf{U}\mathbf{H}\mathbf{U}^\top)\\ &\mathbf{U}\mathbf{H}^\top\mathbf{U}^\top) + \alpha(\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2)\end{aligned} \tag{A.1}$$

the partial derivations of $\mathbf{U}$ and $\mathbf{H}$ with respective to $\mathcal{J}$ can be obtained from $\mathcal{L}$ are

$$\begin{aligned}&\frac{1}{2}\frac{\partial \mathcal{J}}{\partial \mathbf{U}} = \frac{1}{2}\frac{\partial \mathcal{L}}{\partial \mathbf{U}} =\\ &-(\mathbf{W} \odot \mathbf{W} \odot \mathbf{A})\mathbf{U}\mathbf{H}^\top - (\mathbf{W} \odot \mathbf{W} \odot \mathbf{A})^\top\mathbf{U}\mathbf{H} + \alpha\mathbf{U}\\ &+(\mathbf{W} \odot \mathbf{W} \odot \mathbf{U}\mathbf{H}\mathbf{U}^\top)\mathbf{U}\mathbf{H}^\top + (\mathbf{W} \odot \mathbf{W} \odot \mathbf{U}\mathbf{H}\mathbf{U}^\top)^\top\mathbf{U}\mathbf{H},\\ &\frac{1}{2}\frac{\partial \mathcal{J}}{\partial \mathbf{H}} = \frac{1}{2}\frac{\partial \mathcal{L}}{\partial \mathbf{H}} =\\ &-\mathbf{U}^\top(\mathbf{W} \odot \mathbf{W} \odot \mathbf{A})\mathbf{U} + \mathbf{U}^\top(\mathbf{W} \odot \mathbf{W} \odot \mathbf{U}\mathbf{H}\mathbf{U}^\top)\mathbf{U} + \alpha\mathbf{H}\end{aligned} \tag{A.2}$$

Set $\mathbf{B} = \mathbf{G} - \mathbf{U}\mathbf{H}\mathbf{U}^\top$ and let $\mathcal{L}_r$ contain terms related to $\mathbf{r}$ in $\mathcal{J}$, which can be rewritten as,

$$\begin{aligned}\mathcal{L}_r = Tr(&-2\lambda(\mathbf{W} \odot \mathbf{W} \odot \mathbf{B})\mathbf{r}\mathbf{r}^\top\\ &+\lambda^2(\mathbf{W} \odot \mathbf{W} \odot \mathbf{r}\mathbf{r}^\top)\mathbf{r}\mathbf{r}^\top) + \alpha\|\mathbf{r}\|_2^2\end{aligned} \tag{A.3}$$

then the partial derivation of $\mathbf{r}$ with respect to $\mathcal{J}$ is

$$\begin{aligned}&\frac{1}{2}\frac{\partial \mathcal{J}}{\partial \mathbf{r}} = \frac{1}{2}\frac{\partial \mathcal{L}_r}{\partial \mathbf{r}}\\ &-\lambda(\mathbf{W} \odot \mathbf{W} \odot \mathbf{B})\mathbf{r} - \lambda(\mathbf{W} \odot \mathbf{W} \odot \mathbf{B})^\top\mathbf{r} + \alpha\mathbf{r}\\ &+\lambda^2(\mathbf{W} \odot \mathbf{W} \odot \mathbf{r}\mathbf{r}^\top)\mathbf{r} + \lambda^2(\mathbf{W} \odot \mathbf{W} \odot \mathbf{r}\mathbf{r}^\top)^\top\mathbf{r}\end{aligned} \tag{A.4}$$

With the partial derivations of $\mathbf{U}$, $\mathbf{H}$, and $\mathbf{r}$, a optimal solution of the objective function in Eq. (4.9) can be obtained through a gradient decent optimization method as shown in Algorithm 3.

Next we briefly review Algorithm 3. In line 1, we construct the trust and pseudo distrust relation matrix $\mathbf{F}$ and its weight matrix $\mathbf{W}$ from user-user trust relations $\mathbf{T}$, user-review authorship relations $\mathbf{P}$, and user-review helpfulness ratings $\mathbf{R}$. From line 3 to line 8, we update $\mathbf{U}$, $\mathbf{H}$ and $\mathbf{r}$ until convergence where γ_u, γ_h and γ_r are learning steps, which are chosen to satisfy Goldstein Conditions [60]. After learning the user preference matrix $\mathbf{U}$, $\mathbf{H}$ and $\mathbf{r}$ via Algorithm 3, the reconstructed trust and distrust matrix is $\hat{\mathbf{F}} = \mathbf{U}\mathbf{H}\mathbf{U}^\top + \lambda\mathbf{r}\mathbf{r}^\top$. Finally we predict pairs $\langle u_i, u_j \rangle$ whose $sign(\hat{\mathbf{F}}_{ij}) = -1$ as a distrust relation with confidence $|\hat{\mathbf{F}}_{ij}|$.

Algorithm 3 The Proposed Framework dTrust.

Input : User-user trust relations $\mathbf{T}$, user-review authorship relations $\mathbf{P}$, user-review helpfulness ratings $\mathbf{R}$, $\{d, \lambda\}$.

Output : A ranking list of pairs of users.

```
Construct $\mathbf{W}$ and $\mathbf{F}$ from $\mathbf{T}$, $\mathbf{P}$, and $\mathbf{R}$
Initialize $\mathbf{U}$, $\mathbf{H}$ and $\mathbf{r}$ randomly
while Not convergent do
    Calculate $\frac{\partial \mathcal{J}}{\partial \mathbf{U}}$, $\frac{\partial \mathcal{J}}{\partial \mathbf{H}}$ and $\frac{\partial \mathcal{J}}{\partial \mathbf{r}}$
    Update $\mathbf{U} \leftarrow \mathbf{U} - \gamma_u \frac{\partial \mathcal{J}}{\partial \mathbf{U}}$
    Update $\mathbf{H} \leftarrow \mathbf{H} - \gamma_h \frac{\partial \mathcal{J}}{\partial \mathbf{H}}$
    Update $\mathbf{r} \leftarrow \mathbf{r} - \gamma_r \frac{\partial \mathcal{J}}{\partial \mathbf{r}}$
end while
Set $\hat{\mathbf{F}} = \mathbf{U}\mathbf{H}\mathbf{U}^\top + \lambda \mathbf{r}\mathbf{r}^\top$
Set $\mathcal{D} = \{\langle u_i, u_j \rangle | sign(\hat{\mathbf{F}}_{ij}) = -1\}$
Ranking pairs of users in $\mathcal{D}$ (e.g.,$\langle u_i, u_j \rangle$) according to $|\hat{\mathbf{F}}|$ (e.g., $|\hat{\mathbf{F}}_{ij}$—) in a
    descending order
```

APPENDIX B

AN OPTIMIZATION ALGORITHM FOR THE SUPERVISED FRAMEWORK

We solve the optimization problem in Eq. (4.14) based on the dual form [4]. The classical representer theorem states that the solution to this minimization problem of Eq.(4.14) exists in $\mathcal{H}_K$ and can be written as follows:

$$\mathbf{w}^* = \sum_i \alpha_i K(\mathbf{x}_i, \mathbf{x}) \tag{B.1}$$

Eq.(4.14) can be rewritten as follows:

$$\begin{aligned} \min_{\alpha,b,\epsilon} \quad & \frac{1}{2}\alpha^\top \mathbf{K}\alpha + C_p \sum_{u_i \in PS} \epsilon_i + C_n \sum_{u_j \in NS} c_j \epsilon_j + \frac{C_b}{2}\alpha^\top \mathbf{K}\mathcal{L}\mathbf{K}\alpha \\ s.t. \qquad & y_i(\sum_k \alpha_k K(\mathbf{x}_k, \mathbf{x}_i) + b) \geq 1 - \epsilon_i, \quad u_i \in PS \\ & y_j(\sum_k \alpha_k K(\mathbf{x}_k, \mathbf{x}_j) + b) \geq 1 - \epsilon_j, \quad u_j \in NS \\ & \epsilon_i \geq 0, \ \epsilon_j \geq 0 \end{aligned} \tag{B.2}$$

where $\mathbf{K}$ is the Gram matrix over all samples.

We define s_i for x_i as follows:

$$s_i = \begin{cases} C_p & \text{for } x_i \in PS, \\ C_n c_i & \text{for } x_i \in NS. \end{cases} \tag{B.3}$$

After the introduction of two sets of multipliers β and γ, the Lagrangian function of Eq.(B.2) is as follows:

$$\begin{aligned} L(\mathbf{w}, b, \epsilon, \alpha, \gamma) = \frac{1}{2}\alpha^\top(\mathbf{K} + C_b \mathbf{K}\mathcal{L}\mathbf{K})\alpha + \sum_{i=1}^{l} s_i \epsilon_i \\ - \sum_{i=1}^{l} \beta_i [y_i(\sum_k \alpha_k K(\mathbf{x}_k, \mathbf{x}_i) + b) - 1 + \epsilon_i] - \sum_{i=1}^{l} \gamma_i \epsilon_i \end{aligned} \tag{B.4}$$

where β and γ are Lagrange multipliers.

To obtain the dual representation, we set

$$\begin{aligned} \frac{\partial L}{\partial b} &= 0 \Rightarrow \sum_{i=1}^{l} \beta_i y_i = 0 \\ \frac{\partial L}{\partial \epsilon_i} &= 0 \Rightarrow s_i - \beta_i - \gamma_i = 0 \Rightarrow 0 \leq \beta_i \leq s_i \end{aligned} \tag{B.5}$$

With Eq. (B.5), we can rewrite the Lagrangian as a function of only α and β as follows:

$$L(\alpha, \beta) = \frac{1}{2}\alpha^\top(\mathbf{K} + C_b \mathbf{K}\mathcal{L}\mathbf{K})\alpha - \alpha^\top \mathbf{K}\mathbf{J}^\top \mathbf{Y}\beta + \sum_{i=1}^{l} \beta_i \tag{B.6}$$

in Eq. (B.6), $\mathbf{J} = [\mathbf{I}\ \mathbf{0}]$ where $\mathbf{I}$ is an $l \times l$ identity matrix and $\mathbf{0}$ is a $l \times \mu$ rectangular matrix with all zeros, and $\mathbf{Y}$ is a $l \times l$ diagonal matrix composed by labels of samples in PS and NS.

By setting $\frac{\partial L}{\partial \alpha} = 0$, we obtain

$$\alpha = (\mathbf{I} + C_b \mathbf{K}\mathcal{L})^{-1} \mathbf{J}^\top \mathbf{Y} \beta \tag{B.7}$$

After substituting back in the Lagrangian function, we obtain the dual problem as a quadratic programming problem:

$$\begin{aligned} \max_{\beta} \quad & \sum_{i=1}^{l} \beta_i - \frac{1}{2} \beta^\top \mathbf{Q} \beta \\ s.t. \quad & \sum_{i=1}^{l} \beta_i y_i = 0 \\ & 0 \leq \beta_i \leq s_i \end{aligned} \tag{B.8}$$

where $\mathbf{Q}$ is defined as follows:

$$\mathbf{Q} = \mathbf{Y}\mathbf{J}\mathbf{K}(\mathbf{I} + C_b \mathbf{K}\mathcal{L})^{-1} \mathbf{J}^\top \mathbf{Y} \tag{B.9}$$

APPENDIX C

AN OPTIMIZATION ALGORITHM FOR NCSSN

Let $\mathcal{L}$ be the Lagrangian function as:

$$\begin{aligned}\mathcal{L} = f(\mathbf{H}^t, \mathbf{H}^d, \mathbf{U}, \mathbf{W}) - Tr((\Lambda^U)^\top U) \\ - Tr((\Lambda^{H^t})^\top \mathbf{H}^t) - Tr((\Lambda^{H^d})^\top \mathbf{H}^d)\end{aligned} \tag{C.1}$$

where $f(\mathbf{H}^t, \mathbf{H}^d, \mathbf{U}, \mathbf{W})$ is the objective function of Eq. (5.11). The notations Λ^U, Λ^{H^t} and Λ^{H^d} are the Lagrangian multipliers for non-negativity of $\mathbf{U}$, $\mathbf{H}^t$ and $\mathbf{H}^d$.

To compute $\mathbf{U}$*, we fix* $\mathbf{H}^t$*,* $\mathbf{H}^d$ *and* $\mathbf{W}$. The derivative of $\mathcal{L}$ with respect to $\mathbf{U}$ is as follows:

$$\frac{\partial \mathcal{L}}{\partial \mathbf{U}} = \mathbf{E} - \mathbf{B} - \Lambda^U \tag{C.2}$$

where the matrices $\mathbf{B}$ and $\mathbf{E}$ are defined as follows:

$$\begin{aligned}&\mathbf{B} = (\mathbf{T})^\top \mathbf{U}\mathbf{H}^t + \mathbf{T}\mathbf{U}(\mathbf{H}^t)^\top + \alpha\big((\mathbf{C}\mathbf{Y}\mathbf{W}^\top)^+ \\ &+ (\mathbf{C}\mathbf{U}\mathbf{W}\mathbf{W}^\top)^-\big) + \beta((\mathbf{D})^\top \mathbf{U}\mathbf{H}^d + \mathbf{D}\mathbf{U}(\mathbf{H}^d)^\top) \\ &+ \beta\big(\sum_{\langle i,j,k\rangle \in \mathcal{S}} f_{ijk}(\mathbf{M}^{ijk}\mathbf{U} + \mathbf{U}(\mathbf{M}^{ijk})^\top)\big)^- \\ &\mathbf{E} = \mathbf{U}(\mathbf{H}^t)^\top \mathbf{U}^\top \mathbf{U}\mathbf{H}^t + \mathbf{U}\mathbf{H}^t\mathbf{U}^\top\mathbf{U}(\mathbf{H}^t)^\top + \alpha\big((\mathbf{C}\mathbf{U}\mathbf{W}\mathbf{W}^\top)^+ \\ &+ (\mathbf{C}\mathbf{Y}\mathbf{W}^\top)^-\big) + \beta(\mathbf{U}(\mathbf{H}^d)^\top\mathbf{U}^\top\mathbf{U}\mathbf{H}^d + \mathbf{U}\mathbf{H}_d\mathbf{U}^\top\mathbf{U}(\mathbf{H}^d)) \\ &+ \beta\big(\sum_{\langle i,j,k\rangle \in \mathcal{S}} f_{ijk}(\mathbf{M}^{ijk}\mathbf{U} + \mathbf{U}(\mathbf{M}^{ijk})^\top)\big)^+ + \lambda\mathbf{U}\end{aligned} \tag{C.3}$$

where for any matrix $\mathbf{X}$, $(\mathbf{X})^+$ and $(\mathbf{X})^-$ denote the positive and negative parts of $\mathbf{X}$, respectively.

Setting $\frac{\partial \mathcal{L}}{\partial \mathbf{U}} = 0$ and using the KKT complementary condition $\mathbf{U}_{ij}\Lambda^U_{ij} = 0$, we can derive the update rule for $\mathbf{U}$ as follows:

$$\mathbf{U}_{ij} \leftarrow \mathbf{U}_{ij}\sqrt{\frac{\mathbf{B}_{ij}}{\mathbf{E}_{ij}}} \tag{C.4}$$

To compute $\mathbf{H}^t$*, we fix* $\mathbf{U}$*,* $\mathbf{H}^d$ *and* $\mathbf{W}$. The derivative of $\mathcal{L}$ with respect to $\mathbf{H}^t$ is as follows:

$$\frac{\partial \mathcal{L}}{\partial \mathbf{H}^t} = \mathbf{U}^\top\mathbf{U}\mathbf{H}^t\mathbf{U}^\top\mathbf{U} + \lambda\mathbf{H}^t - \mathbf{U}^\top\mathbf{T}\mathbf{U} - \Lambda^{H^t} \tag{C.5}$$

Setting $\frac{\partial \mathcal{L}}{\partial \mathbf{H}^t} = 0$ and using the KKT complementary condition $\mathbf{H}^t_{ij}\Lambda^{H^t}_{ij} = 0$, we can get the update rule for $\mathbf{H}^t$ as follows:

$$\mathbf{H}^t_{ij} \leftarrow \mathbf{H}^t_{ij}\sqrt{\frac{[\mathbf{U}^\top\mathbf{T}\mathbf{U}]_{ij}}{[\mathbf{U}^\top\mathbf{U}\mathbf{H}^t\mathbf{U}^\top\mathbf{U} + \lambda\mathbf{H}^t]_{ij}}} \tag{C.6}$$

Similarly, we can obtain the update rule for $\mathbf{H}^d$ as follows:

$$\mathbf{H}^d_{ij} \leftarrow \mathbf{H}^d_{ij}\sqrt{\frac{[\mathbf{U}^\top \mathbf{D}\mathbf{U}] - ij}{[\mathbf{U}^\top \mathbf{U}\mathbf{H}^d\mathbf{U}^\top \mathbf{U} + \lambda \mathbf{H}^d]_{ij}}} \tag{C.7}$$

Setting $\frac{\partial \mathcal{L}}{\partial \mathbf{W}} = 0$, we obtain the following:

$$\mathbf{W} = (\mathbf{U}^\top \mathbf{C}\mathbf{U} + \lambda \mathbf{I})^{-1}\mathbf{U}^\top \mathbf{C}\mathbf{Y} \tag{C.8}$$

With update rules for $\mathbf{U}$, $\mathbf{H}^t$, $\mathbf{H}^d$ and $\mathbf{W}$, the detailed algorithm for NCSNN is shown in Algorithm 4. Next, we give a brief description of Algorithm 4. In line 1, we construct the set $\mathcal{S}$ and then we initialize $\mathbf{U}$, $\mathbf{H}^t$, $\mathbf{H}^d$ and $\mathbf{W}$ randomly. From line 3 to line 10, we update $\mathbf{U}$, $\mathbf{H}^t$, $\mathbf{H}^d$ and $\mathbf{W}$ according to update rules in Eqs. (C.4), (C.6), (C.7) and (C.8).

Algorithm 4 The Node Classification Framework in Trust/Distrust Networks.

```
Input: $\mathbf{T}$, $\mathbf{D}$, $\mathbf{Y}$ and $\{\alpha, \beta, \lambda\}$
Output: A linear classifier $\mathbf{W}$ and the user latent factor matrix $\mathbf{U}$
 1: Construct $\mathcal{S}$ as $\mathcal{S} = \{\langle i, j, k\rangle | \mathbf{T}_{ij} = 1 \wedge \mathbf{D}_{ik} = 1\}$
 2: Initialize $\mathbf{U}$, $\mathbf{H}^t$, $\mathbf{H}^d$ and $\mathbf{W}$ randomly
 3: while Not convergent do
 4:   for $\langle i, j, k\rangle \in \mathcal{S}$ do
 5:     Construct $f_{ijk}$ and $\mathbf{M}^{ijk}$
 6:   end for
 7:   Construct $\mathbf{B}$ and $\mathbf{E}$ as Eq. (C.3)
 8:   for $i$ from 1 to $N$ do
 9:     for $j$ from 1 to $K$ do
10:       $\mathbf{U}_{ij} \leftarrow \mathbf{U}_{ij}\sqrt{\frac{\mathbf{B}_{ij}}{\mathbf{E}_{ij}}}$
11:     end for
12:   end for
13:   for $i$ from 1 to $K$ do
14:     for $j$ from 1 to $K$ do
15:       $\mathbf{H}^t_{ij} \leftarrow \mathbf{H}^t_{ij}\sqrt{\frac{[\mathbf{U}^\top \mathbf{T}\mathbf{U}]_{ij}}{[\mathbf{U}^\top \mathbf{U}\mathbf{H}^t\mathbf{U}^\top \mathbf{U} + \lambda \mathbf{H}^t]_{ij}}}$
16:       $\mathbf{H}^d_{ij} \leftarrow \mathbf{H}^d_{ij}\sqrt{\frac{[\mathbf{U}^\top \mathbf{D}\mathbf{U}] - ij}{[\mathbf{U}^\top \mathbf{U}\mathbf{H}^d\mathbf{U}^\top \mathbf{U} + \lambda \mathbf{H}^d]_{ij}}}$
17:     end for
18:   end for
19:   $\mathbf{W} = (\mathbf{U}^\top \mathbf{C}\mathbf{U} + \lambda \mathbf{I})^{-1}\mathbf{U}^\top \mathbf{C}\mathbf{Y}$
20: end while
```

APPENDIX D

AN OPTIMIZATION ALGORITHM FOR RECSSN

We define $\mathbf{M}_i^k$ at the k-th iteration for u_i as follows:

$$\mathbf{M}_i^k = \quad \left\{ \begin{array}{ll} 1 & \|\mathbf{U}_i - \bar{\mathbf{U}}_i^p\|_2^2 - \|\mathbf{U}_i - \bar{\mathbf{U}}_i^n\|_2^2 > 0 \\ 0 & \text{otherwise} \end{array} \right. . \tag{D.1}$$

Then, we use $\mathcal{J}$ to denote the objective function of Eq. (5.21) in the k-th iteration as follows:

$$\begin{aligned} \mathcal{J} = & \sum_{i=1}^{N}\sum_{j=1}^{m} g(\mathbf{W}_{ij}, \mathbf{w}_i)\|(\mathbf{R}_{ij} - \mathbf{U}_i\mathbf{V}_j^\top)\|_2^2 \\ & + \alpha(\sum_{i=1}^{N}\|\mathbf{U}\|_2^2 + \sum_{j=1}^{m}\|\mathbf{V}_j\|_2^2) + \beta\sum_{i=1}^{N}\mathbf{M}_i^k(\\ & \|\mathbf{U}_i - \frac{\sum_{u_j\in\mathcal{P}_i}\mathbf{S}_{ij}\mathbf{U}_j}{\sum_{u_j\in\mathcal{P}_i}\mathbf{S}_{ij}}\|_2^2 - \|\mathbf{U}_i - \frac{\sum_{u_j\in\mathcal{N}_i}\mathbf{S}_{ij}\mathbf{U}_j}{\sum_{u_j\in\mathcal{N}_i}\mathbf{S}_{ij}}\|_2^2) \end{aligned} \tag{D.2}$$

The derivatives of $\mathcal{J}$ with respect to $\mathbf{U}_i$ and $\mathbf{V}_j$ are as follows:

$$\begin{aligned} \frac{\partial\mathcal{J}}{\partial\mathbf{U}_i} = & -2\sum_j g(\mathbf{W}_{ij}, \mathbf{w}_i)(\mathbf{R}_{ij} - \mathbf{U}_i\mathbf{V}_j^\top)\mathbf{V}_j + 2\alpha\mathbf{U}_i \\ & + 2\beta\mathbf{M}_i^k(\mathbf{U}_i - \bar{\mathbf{U}}_i^p) - 2\beta\mathbf{M}_i^k(\mathbf{U}_i - \bar{\mathbf{U}}_i^n) \\ & - 2\beta\sum_{u_j\in\mathcal{P}_i}\mathbf{M}_j^k(\mathbf{U}_j - \bar{\mathbf{U}}_j^p)\frac{1}{\sum_{u_j\in\mathcal{P}_i}\mathbf{S}_{ji}} \\ & + 2\beta\sum_{u_j\in\mathcal{N}_i}\mathbf{M}_j^k(\mathbf{U}_j - \bar{\mathbf{U}}_j^n)\frac{1}{\sum_{u_j\in\mathcal{N}_i}\mathbf{S}_{ji}} \\ \frac{\partial\mathcal{J}}{\partial\mathbf{V}_j} = & -2\sum_i g(\mathbf{W}_{ij}, \mathbf{w}_i)(\mathbf{R}_{ij} - \mathbf{U}_i\mathbf{V}_j^\top)\mathbf{U}_i + 2\alpha\mathbf{V}_j \end{aligned} \tag{D.3}$$

The detailed algorithm is shown in Algorithm 5. In Algorithm 5, γ_u and γ_v are learning steps, which are chosen to satisfy Goldstein Conditions [60]. Next, we briefly discuss the algorithm. In line 1, we initialize latent factors of users $\mathbf{U}$ and items $\mathbf{V}$ randomly. In each iteration, we calculate $\bar{\mathbf{U}}_i^p$, $\bar{\mathbf{U}}_i^n$ and $\mathbf{M}_i^k$ for u_i from line 3 to line 6. From line 7 to line 9, we update $\mathbf{U}$ and $\mathbf{V}$ using aforementioned update rules.

Algorithm 5 The Proposed Recommendation Framework RecSSN with Signed Social Networks.

Input: The rating information $\mathbf{R}$, trust links $\mathbf{T}$, negative links $\mathbf{D}$, the number of latent factors K and β

Output: The user preference matrix $\mathbf{U}$ and the item characteristic matrix $\mathbf{V}$

```
Initialize $\mathbf{U}$ and $\mathbf{V}$ randomly and set $k = 1$
while Not convergent do
    for $i = 1 : N$ do
        Calculate $\bar{\mathbf{U}}_i^p$ and $\bar{\mathbf{U}}_i^n$ according to Eq. (5.18)
        Calculate $\mathbf{M}_i^k$ according to Eq. (D.1)
    end for
    Calculate $\frac{\partial \mathcal{J}}{\partial \mathbf{U}}$ and $\frac{\partial \mathcal{J}}{\partial \mathbf{V}}$
    Update $\mathbf{U} \leftarrow \mathbf{U} - \gamma_u \frac{\partial \mathcal{J}}{\partial \mathbf{U}}$
    Update $\mathbf{V} \leftarrow \mathbf{V} - \gamma_v \frac{\partial \mathcal{J}}{\partial \mathbf{V}}$
    k = k + 1
end while
```

BIOGRAPHICAL SKETCH

Jiliang Tang is a senior PhD student of Computer Science and Engineering at Arizona State University. He obtained his Master degree in Computer Science and Bachelor degree in Software Engineering at Beijing Institute of Technology in 2008 and 2010, respectively. He was awarded the 2014 ASU Presidents Award for Innovation, Best Paper Shortlist in WSDM13, the 3rd Place Dedicated Task 2 Next Location Prediction of Nokia Mobile Data Challenge 2012, University Graduate Fellowship, and various Student Travel Awards and Scholarships. His research interests are in computing with online trust and distrust, recommendation, mining social media data, data mining and feature selection. He copresented three tutorials in KDD2014, WWW2014 and Recsys2014, and has published more than 50 innovative works in highly ranked journals and top conference proceedings such as IEEE TKDE, ACM TKDD, DMKD, ACM SIGKDD, SIGIR, WWW, WSDM, SDM, ICDM, IJCAI, AAAI, and CIKM. He also worked as a research intern at Yahoo!Labs in 2013 and IBM T.J. Watson Research in 2014.